



**Diogo José
Simões Martins**

Emissor baseado em SDR para comunicações DSRC



**Diogo José
Simões Martins**

Emissor baseado em SDR para comunicações DSRC

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Electrónica e Telecomunicações, realizada sob a orientação científica do Doutor João Nuno Matos e do Doutor Arnaldo Silva Rodrigues de Oliveira, Professores do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

o júri / the jury

presidente / president

Prof. Doutor Nuno Miguel Gonçalves Borges de Carvalho

Professor Associado com agregação da Universidade de Aveiro

vogais / examiners committee

Prof. Doutor José Carlos dos Santos Alves

Professor Associado do Departamento de Engenharia Eletrotécnica e Computadores da Faculdade de Engenharia da Universidade do Porto

Prof. Doutor João Nuno Pimentel da Silva Matos

Professor Associado da Universidade de Aveiro (Orientador)

Prof. Doutor Arnaldo Silva Rodrigues de Oliveira

Professor Auxiliar da Universidade de Aveiro (Coorientador)

agradecimentos

Deixo aqui os meus mais profundos agradecimentos a todas as pessoas, que de uma forma ou de outra, tornaram este trabalho possível e me ajudaram no seu desenvolvimento e conclusão.

São elas:

Orientadores e colegas de trabalho, com particular destaque para o Ricardo Matos, pela sua disponibilidade, ajuda e orientação ao longo de todo o trabalho;

Família e amigos.

Um agradecimento especial ao Rex e ao Rufus.

Palavras-chave

EFC, DSRC, SDR

Resumo

Esta dissertação insere-se na temática das comunicações DSRC a 5.8 GHz, para sistemas de cobrança eletrónica de taxas variadas (exemplo: circulação em autoestradas).

Um sistema DSRC é constituído por duas unidades, uma fixa colocada na via (RSU) e uma móvel colocada no veículo (OBU). O objetivo desta dissertação é o desenvolvimento do emissor que irá ser englobado na RSU, recorrendo à tecnologia denominada por *software defined radio*. Em sistemas SDR faz-se a modelação em domínio digital de componentes analógicos, o que confere vantagens tanto para concessionárias como para fabricantes.

O emissor projetado é composto por duas partes: uma completamente digital, implementada numa FPGA e responsável pelo tratamento efetuado em banda base; E outra referente ao *front-end*, responsável pela conversão digital-analógico e por todo o tratamento em radiofrequência. A implementação do sistema na FPGA foi a parte de maior atenção ao longo do trabalho, ressaltando-se também algum foco para a parte relativa ao conversor digital-analógico e filtro analógico.

A abordagem que se seguiu passou pelo estudo das normas que regem as comunicações DSRC 5.8 GHz, importantes do ponto de vista do trabalho (camada física e camada MAC), de maneira a melhor estruturar a arquitetura do sistema. Feito esse estudo procedeu-se de seguida à execução de uma série de simulações, por forma a escolher qual a melhor solução a optar para a implementação física do sistema (meta seguinte do trabalho).

Por fim efetuaram-se testes visando a validação do dispositivo construído. Os testes recaíram sobre o cumprimento ou não das máscaras espectrais referidas na norma referente à camada física. Recolha de diagramas de olho por forma a qualificar o sinal enviado. E execução de uma comunicação entre o emissor criado e um equipamento comercial, referente à OBU.

Keywords

EFC, DSRC, SDR

Abstract

This dissertation falls in the theme of the 5.8 GHz DSRC communications for electronic fee collection systems (eg: highways circulation).

A DSRC system is formed by two units, a fixed one, placed in a route (RSU) and a mobile one, placed on a vehicle (OBU). The goal of this dissertation is the development of the transmitter which will be enclosed in RSU, using technology called software defined radio. The SDR allows the modeling of analog components in digital domain, which brings advantages to both manufacturers and utilities.

The transmitter designed is composed by two parts: one completely digital, implemented on a FPGA and responsible for the base band treatment; other concerning the front-end, responsible for the digital to analog conversion and for all the treatment in radiofrequency. The part concerning the FPGA implementation was object of a greater attention along the work, although the digital-analog conversion and the analog filter also play a important role.

The approach followed passes through the study of the rules governing 5.8 GHz DSRC communications, important from the standpoint of work (physical layer and MAC layer), in order to better structure the system architecture. Done this it was made a several simulations in order to choose the best solution to opt for the physical implementation of the system (next goal of the work).

Lastly it will be made tests in order to validate the constructed transmitter. It will be tested the fulfillment of spectral masks and it will be used eye diagrams to evaluate the quality and performance of the system output. Ultimately it will be made a communication between the constructed transmitter and commercial equipment, corresponding to the mobile unit placed at the vehicle (OBU).

Conteúdo

Conteúdo	i
Lista de Figuras	iii
Lista de Tabelas	v
Acrónimos	vii
1 Introdução	1
1.1 Enquadramento	1
1.2 Motivação	2
1.3 Objetivos	2
1.4 Estrutura da Dissertação	3
2 Estado da Arte	5
2.1 Sistemas EFC	5
2.2 DSRC	6
2.2.1 Arquitetura	7
2.2.2 Protocolo	8
2.2.3 Camada física	9
2.2.4 Camada de ligação de dados	13
2.3 Sistemas <i>software defined radio</i>	14
2.3.1 Utilização de FPGAs em SDR	15
2.3.2 Arquitetura do emissor com base em SDR	15
2.3.3 Conversor digital-analógico	16
2.3.4 Filtros em SDR	17
2.3.5 Outros Projetos baseados em SDR	18
3 Modelação e simulação do emissor	21
3.1 Introdução	21
3.2 Modelação	21
3.2.1 Arquitetura do sistema	21
3.2.2 Modelo implementado no <i>Matlab</i>	22
3.2.3 Estrutura do modelo do <i>Matlab</i>	23
3.3 Simulação	25
3.3.1 Objetivos	26
3.3.2 Filtros simulados no <i>Matlab</i>	26

3.3.3	Parâmetros de entrada da simulação	29
3.3.4	Cumprimento das máscaras espectrais sem DAC	29
3.3.5	Cumprimento das máscaras espectrais com DAC	33
3.3.6	Conclusões	36
4	Implementação do emissor	39
4.1	Introdução	39
4.2	Arquitetura do emissor	39
4.3	Modelo digital	45
4.4	Front-End Analógico	47
4.4.1	Conversor Digital-Analógico	48
4.4.2	Filtro Analógico	49
5	Validação experimental e resultados	51
5.1	Introdução	51
5.2	Máscaras espectrais	51
5.2.1	Testes e resultados	52
5.3	Diagramas de olho	59
5.3.1	Testes e resultados	60
5.4	Comunicação entre RSU e OBU	63
5.4.1	Testes e resultados	63
6	Conclusões e trabalho futuro	69
6.1	Conclusões	69
6.2	Trabalho futuro	70
	Apêndices	71
A	Código MatLab do emissor	73
B	Front-End	83
	Bibliografia	87

Lista de Figuras

1.1	Exemplo de um sistema EFC	2
2.1	Modelo OSI colapsado típico em sistemas <i>embedded</i> distribuídos	7
2.2	Exemplo de um sistema EFC	7
2.3	Arquitetura de um sistema DSRC (pormenorizada)	8
2.4	Exemplo de uma codificação FM0	10
2.5	Máscaras espectrais (LDR)	11
2.6	Máscaras espectrais (MDR)	12
2.7	Portadoras e sub-portadoras de uma comunicação DSRC	13
2.8	Exemplo de uma <i>flag</i> de sinalização	13
2.9	Emissor baseado na tecnologia SDR	16
3.1	Diagrama geral de blocos do emissor	22
3.2	Sistema modelado	22
3.3	Diagrama de fluxo do modelo do <i>Matlab</i>	23
3.4	Respostas em frequência e fase dos filtros ($F_a = 20$ MHz)	27
3.5	Resposta em frequência de um filtro FIR e uma cascata de 4 filtros FIR	31
3.6	Diagramas temporais relativos ao filtro FIR	32
3.7	Diagramas temporais relativos ao filtro Butterworth	32
3.8	Diagramas temporais relativos ao filtro elíptico	33
3.9	Ruido de quantificação (FIR)	34
3.10	Ruido de quantificação (Butterworth)	35
3.11	Ruido de quantificação (Elíptico)	35
4.1	Diagrama de blocos do emissor referente ao interior da FPGA	40
4.2	Exemplo de uma sequência pseudo-aleatória	42
4.3	Shift-register implementado para criar a sequência pseudo-aleatória	42
4.4	Janela de configuração (<i>FDA tool</i>)	44
4.5	Placa de desenvolvimento	46
4.6	Modelo do conversor digital-analógico	48
4.7	Espectro do filtro analógico	50
5.1	Equipamentos envolvidos na aquisição de resultados	52
5.2	Placa de desenvolvimento, conversor Digital-Analógico e Filtro Analógico	53
5.3	Diagrama de fluxo (medidas de radiação)	54
5.4	Variação do número de bits de quantificação da DAC	56
5.5	Variação da frequência de amostragem	57

5.6	Aplicação do filtro analógico	58
5.7	Características de um diagrama de olho	59
5.8	Diagrama de fluxo (diagramas de olho)	60
5.9	Diagrama de olho (sinal 20MHz, 12 bits)	61
5.10	a)-Diagrama de olho (sinal 5MHz, 12 bits); b)-Diagrama de olho (sinal 10MHz, 12 bits)	61
5.11	a)-Diagrama de olho (sinal 20MHz, 10 bits); b)-Diagrama de olho (sinal 20MHz, 8 bits)	62
5.12	a)-Diagrama de olho (Filtro Analógico (6dB)); b)-Diagrama de olho (Filtro Analógico (10dB))	63
5.13	Teste de uma comunicação com uma OBU	64
5.14	Diagrama de fluxo (comunicação entre RSU e OBU)	65
5.15	a)-Espectro do sinal a 5,8 GHz; b)-Sinal enviado pela RSU (em cima) e desmodulado pela OBU (em baixo)	65
5.16	Preâmbulo da trama de resposta da OBU	66
5.17	<i>Flag</i> de início da trama de resposta da OBU	67
5.18	<i>Flag</i> de fim da trama de resposta da OBU	67
5.19	Exemplo de um trecho da trama de resposta da OBU	68
B.1	Esquema do PCB a ser construído	85
B.2	PCB construído (top)	86
B.3	PCB construído (bottom)	86

Lista de Tabelas

2.1	Máscaras espectrais e suas diferentes classes	11
2.2	Máscaras espectrais para todo espectro	12
3.1	Máscaras espectrais e suas diferentes classes (valores normalizados)	26
3.2	Comparação entre filtros IIR (elíptico e Butterworth)	30
3.3	Variação da frequência de amostragem num filtro FIR	30
3.4	SQNR	36
4.1	Recursos, da FPGA, utilizados	47
5.1	Frequências e RBW para o teste das máscaras espectrais do emissor	54

Acrónimos

ALPR Automatic License Plate Recognition

API Application Programming Interface

ASK Amplitude Shift Keying

BPSK Binary Phase Shift Keying

CEN Comité Européen de Normalização

CRC Cyclic Redundancy Check

DAC Digital to Analog Converter

DC Direct Current

DCM Digital Clock Manager

DSP Digital Signal Processor

DSRC Dedicated Short Range Communications

EDA Electronic Design Automation

EETS European Electronic Toll Service

EFC Electronic Fee Collection

ETSI European Telecommunications Standards Institute

FIR Finite Impulse Response

FPGA Field Programmable Gate Array

GNSS Global Navigation Satellite Systems

GPRS General Packet Radio Service

GPS Global Positioning System

GSM Global System for Mobile

IC Integrated Circuit

IIR Infinite Impulse Response

IP Internet Protocol

ISM Industrial, Scientific and Medical

ISO International Organization for Standardization

ITS Intelligent Transportation Systems

JTAG Joint Test Action Group

JTRS Joint Tactical Radio System

LDR Low Data Rate

LLC Logical Link Control

MAC Medium Access Control

MDR Medium Data Rate

NATO North Atlantic Treaty Organization

NRZI Non Return to Zero Inverted

OBU On Board Unit

OSI Open Systems Interconnection

PCB Printed Circuit Board

PIRE Potência Isotrópica Radiada Equivalente

RBW Resolution BandWidth

ROM Read-Only Memory

RSU Road Side Unit

RTTT Road Transport and Traffic Telematics

SCA Software Communications Architecture

SDR Software Defined Radio

SQNR Signal-to-Quantization-Noise Ratio

TC278 Technical Committee 278

TC204 Technical Committee 204

USB Universal Serial Bus

VHDL VHSIC Hardware Description Language

WG1 WorkGroup 1

WG5 WorkGroup 5

Capítulo 1

Introdução

1.1 Enquadramento

Um sistema de cobrança eletrónica de portagens (EFC) é uma aplicação inserida na área das aplicações de telemática para o tráfego e transporte rodoviário (RTTT). Este sistema foi implementado em Portugal pela Brisa no projeto Via Verde, projeto esse pioneiro em todo mundo, pois Portugal passou a ser o primeiro país a possuir um único serviço de portagem automatizado para todo o país.

Um sistema EFC basicamente consiste em dois dispositivos. Uma unidade móvel (OBU) colocada no veículo e uma unidade fixa (RSU) colocada na via (Figura 1.1). Este sistema é usado para a cobrança automática de determinadas taxas, entre as quais a circulação em autoestradas pagas, parques de estacionamento e bombas de gasolina. A tecnologia que garante a comunicação entre os dispositivos intervenientes é o DSRC. O DSRC é muito usado para a transferência de dados em aplicações RTTT, operando entre as frequências de 5,725GHz e 5,875GHz da banda industrial, científica e médica (ISM).

Os sistemas EFC são projetados para desempenhar tarefas específicas (sistemas *embedded*). Habitualmente em sistemas *embedded* o modelo OSI é reduzido para um número de camadas inferior ao que usualmente se tem. Para cada uma das camadas OSI definidas para o DSRC existem normas europeias EN criadas que ajudam a garantir a compatibilidade e interoperabilidade entre dispositivos diferentes. São elas:

- EN 12253 (*DSRC Physical Layer at 5.8 GHz*) - Caracteriza a camada física e será mais pormenorizadamente discutida neste trabalho;
- EN 12795 (*DSRC Data Link Layer*) - Correspondente à camada de transporte. Apresenta também aspetos importantes para o projeto que se pretende desenvolver;
- EN 12834 (*DSRC Application Layer*) - Define a camada de aplicação. Não será alvo de estudo no trabalho;
- EN 13372 (*DSRC Profiles for RTTT Applications*) - Responsável por definir o perfil das comunicações em RTTT, baseadas na tecnologia DSRC. Também não será alvo de estudo;
- EN 300 674 (*EMR & Test Methods for DSRC*) - Refere-se a testes que os sistemas baseados em DSRC terão de cumprir.

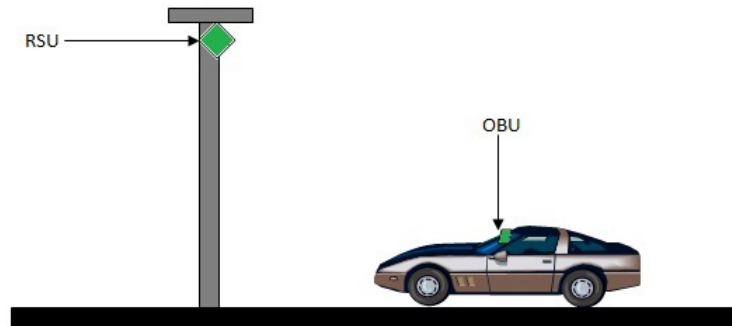


Figura 1.1: Exemplo de um sistema EFC

1.2 Motivação

Como já referido os sistemas EFC são já amplamente utilizados para o pagamento de variados serviços sendo, por isso, uma tecnologia com algum grau de fixação no mercado. Porém as RSUs atuais são aparelhos em que o condicionamento dos sinais de rádio é essencialmente analógico, ou seja, o acondicionamento que é necessário aplicar ao sinal em banda base é feito recorrendo a dispositivos analógicos (componentes físicos com características próprias que não podem ser alteradas). Esta utilização de dispositivos analógicos torna estes sistemas muito inflexíveis, sendo difícil que uma alteração do funcionamento do sistema não implique uma alteração na implementação física do mesmo.

Em sistemas *software defined radio* (SDR - tecnologia discutida mais à frente neste documento) são utilizados processadores especializados no tratamento de sinais, tais como, as FPGAs. Estas permitem a modelação de *hardware* recorrendo a uma linguagem de *software* (VHDL) e, por isso, o uso de FPGAs em sistemas SDR confere a estes uma maior adaptabilidade e reconfigurabilidade em comparação com os sistemas analógicos.

Um RSU baseado na tecnologia SDR apresenta vantagens tanto para o fabricante como para as concessionárias.

Do ponto de vista do fabricante há a vantagem de se reduzir o uso de componentes analógicos diminuindo o custo de fabrico e simplificando a arquitetura do rádio. O uso do SDR facilita ainda o teste de novas soluções e permite uma mais rápida evolução do equipamento.

Já a empresa concessionária terá uma maior facilidade na manutenção dos equipamentos, pois o desgaste dos componentes analógicos é praticamente eliminado. Além disso a implementação de novos serviços é mais simples e rápida, visto ser apenas necessário descarregar o novo software para o equipamento SDR para este passar a funcionar de acordo com as novas especificações.

Face a estes pressupostos e com o crescimento tecnológico na área do *software defined radio*, faz sentido a construção de uma RSU digital.

1.3 Objetivos

A finalidade deste projeto é a criação de um emissor para uma RSU de um sistema EFC. O emissor vai ser construído com base em SDR, ou seja, vai ser praticamente um dispositivo digital, apresentando apenas um *front-end* analógico para tratar da parte de rádiofrequência.

Os vários passos que se pretendem solucionar na execução do projeto serão:

- Estudo da norma EN 12253 relativa à camada física do DSRC e de certos pontos da norma EN 12975 referente à camada de ligação de dados do DSRC;
- Caracterização das especificações do sistema;
- Definição da arquitetura do emissor a implementar;
- Escolha da filtragem a aplicar, de maneira a respeitar as máscaras espectrais;
- Projeto e implementação do módulo que inclui o conversor digital-analógico;
- Projeto e implementação do sistema no interior da FPGA;
- Verificação do desempenho do sistema desenvolvido.

1.4 Estrutura da Dissertação

Além deste capítulo introdutório esta dissertação encontra-se estruturada da seguinte forma:

- Capítulo 2 - Estado da arte: Discussão dos aspetos teóricos importantes para a compreensão do trabalho desenvolvido:
 - Sistemas EFC;
 - Definição e arquitetura do DSRC;
 - Normas referentes à camada física e camada de ligação de dados;
 - Sistemas SDR e consequente referência a FPGAs e DACs.
- Capítulo 3 - Modelação e simulação do emissor: Descrição do modelo a simular. Simulação do mesmo, por forma a escolher quais as melhores soluções a adotar e qual o caminho a seguir. Exposição dos resultados e respetivas conclusões;
- Capítulo 4 - Implementação do emissor: Apresentação e discussão do diagrama de blocos do sistema a ser modelado no interior da FPGA. Referência ao *front-end* utilizado para conversão de dados (DAC);
- Capítulo 5 - Validação experimental e resultados: Apresentação da abordagem seguida para se efetuar os testes e exibição dos resultados obtidos;
- Capítulo 6 - Conclusões e trabalho futuro: Exposição das principais conclusões do trabalho e propostas de trabalho futuro a realizar.

Capítulo 2

Estado da Arte

2.1 Sistemas EFC

Os sistemas EFC apareceram no início da década de 90, sendo que a Noruega foi o país pioneiro na sua implementação. Os equipamentos eram comercializados por diferentes empresas e inoperáveis entre si o que constituía um entrave à sua evolução e implementação, tornando-se por isso necessária a criação de um conjunto de regras para serem seguidas por todas as entidades envolvidas. Visando a resolução deste problema foi lançada a diretiva 2004/52/EC. Esta diretiva pretende garantir a interoperabilidade dos sistemas EFC e uma das medidas aplicadas de maneira a atingir esse fim foi que, após janeiro de 2007, todos os novos sistemas EFC deveriam utilizar uma ou mais das seguintes tecnologias de comunicação:

- Posicionamento por satélite;
- Comunicações móveis (GSM/GPRS);
- Micro-ondas (5,8 GHz).

Estes sistemas subdividem-se em várias tecnologias, dependendo da maneira como é efetuada a comunicação, entre as quais:

- Sistema de navegação global por satélite (GNSS): sistemas que fazem uso do sistema de posicionamento global (GPS) para determinar a posição e distância percorrida por determinado veículo e assim cobrar a taxa correspondente. Estes sistemas têm a vantagem de não necessitarem, ao contrário da maioria dos sistemas EFC, de RSUs. Contudo os custos para o cliente são maiores visto os equipamentos a serem instalados nos veículos serem mais complexos.
- Reconhecimento automático de matrículas (ALPR): têm como base o processamento de imagens provenientes de câmaras de video (OCR). Ou seja, as câmaras usadas nestes equipamentos têm a capacidade de reconhecer as matrículas dos veículos que passam, ficando assim automaticamente registado o veículo nesse local e permitindo a cobrança automática da taxa aplicada.
- DSRC: É de seguida discutido mais pormenorizadamente;
- Sistemas híbridos: DSRC + GNSS e DSRC + ALPR, por exemplo.

2.2 DSRC

Hoje em dia o DSRC não é somente usado em sistemas EFC. Existem já variantes desta tecnologia aplicadas noutras áreas - serviços inteligentes de transporte (ITS) modernos. Estes novos serviços permitem resolver problemas relacionados com o tráfego nas estradas, tais como congestão de tráfego e acidentes, e opera nos 5,9 GHz. O principal objetivo é melhorar a fluidez do tráfego e segurança dos condutores sem haver a necessidade de construir novas estradas. A taxa de transmissão elevada que o DSRC oferece para comunicações de raio reduzido torna-o uma boa escolha [1]. Estes novos sistemas tentam conjugar o DSRC com o protocolo internet (IP), sendo que a distribuição da informação é feita de duas maneiras. Cada veículo pode receber informação aquando da sua passagem por uma RSU, informação essa disponibilizada por instituições governamentais ou departamentos de gestão do tráfego fazendo uso, por exemplo, da internet para passar a informação útil desde a sua localização até à RSU pretendida. Em caso de acidentes a informação relativa a esses acontecimentos poderá ser transmitida diretamente de veículo para veículo e de veículo para RSU. Esta nova possibilidade de comunicação entre OBUs possibilita o aparecimento de outras aplicações. Aplicações que podem ajudar a que haja uma maior disponibilidade de informação para os condutores em relação ao meio envolvente e logo possibilitar uma maior segurança nas estradas e uma condução mais eficiente e segura.

No caso deste trabalho utiliza-se o DSRC criado para aplicações RTTT, de entre as quais a EFC. As aplicações a que se faz referência podem estar relacionadas com o pagamento de taxas de circulação em autoestradas, utilização de parques de estacionamento, abastecimento em bombas de gasolina e ainda gestão de entrada e saída de veículos de determinadas áreas fechadas [2]. Nestes casos as OBUs apenas transmitem informação relativa ao utilizador de maneira a permitir a cobrança da taxa relativa ao serviço em causa e apenas é possível a comunicação entre RSU e OBU. As OBUs utilizadas são do tipo passivo, ou seja, são OBUs que não apresentam oscilador local, conseguindo dessa forma diminuir o hardware necessário tornando-os dispositivos menos complexos e mais económicos [1]. A comunicação entre os dois equipamentos é do tipo *master-slave* em que a RSU é o *master* e a OBU o *slave*. Ou seja, o controlo de todas as fases de uma comunicação é efetuado pela RSU.

O DSRC usado em sistemas EFC opera a 5,8 GHz e possibilita a troca de dados entre um veículo e um determinado posto fixo inserido na estrada.

Estes sistemas são do tipo *embedded* distribuídos. São sistemas projetados para desempenhar tarefas específicas, muito utilizados em sistemas de tempo-real. A sua utilização tem a vantagem de reduzir a complexidade do protocolo e permitir uma comunicação mais rápida. Isto porque, apesar do modelo OSI definir ao todo sete camadas protocolares para se efetuar uma correta comunicação entre dois dispositivos, no caso destes sistemas *embedded* o número de camadas protocolares usadas é reduzido para apenas três (Figura 2.1).

De maneira a permitir que com uma mesma OBU seja possível circular por várias estradas em diferentes países e assegurando em todos a mesma qualidade de serviço foi criado o serviço europeu de portagens electrónicas (EETS). A elaboração de normas, especificações e linhas orientadoras para o apoio dos requisitos da diretiva ficou a cargo das organizações europeias de normalização (CEN, CENELEC e ETSI). Atualmente as normas são desenvolvidas em conjunto por dois grupos, o CEN/TC278/WG1 e o ISO/TC204/WG5.

É importante referir que ao longo do trabalho qualquer referência à tecnologia DSRC relaciona-se com aquela que opera a 5,8 GHz.

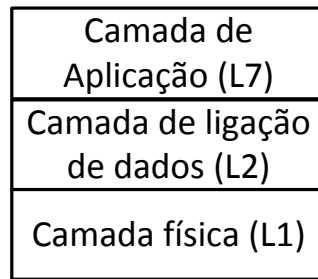


Figura 2.1: Modelo OSI colapsado típico em sistemas *embedded* distribuídos

2.2.1 Arquitetura

Na figura 2.2 é dada a panorâmica geral em relação às partes envolvidas e à sua colocação no espaço. Observa-se que um sistema de comunicações DSRC é constituído por duas partes: uma RSU e uma OBU. A OBU é instalada no veículo enquanto que a RSU é instalada num posto fixo junto à estrada. Em cada um destes equipamentos é feita uma implementação do protocolo relativo a comunicações DSRC permitindo a comunicação entre as unidades.

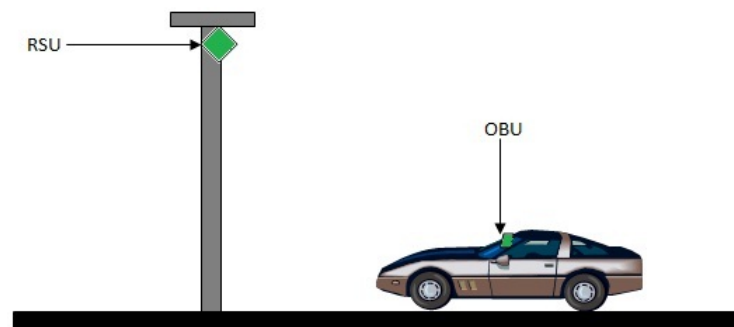


Figura 2.2: Exemplo de um sistema EFC

A figura 2.3 ilustra mais pormenorizadamente a constituição dos dois dispositivos envolvidos na comunicação: RSU e OBU. No caso da RSU e no âmbito deste projeto importa apenas fazer referência ao emissor. Este é constituído por três blocos:

- Microcontrolador: É responsável pela formação da mensagem a enviar e é onde se implementam as configurações relativas às diferentes camadas do protocolo para as comunicações DSRC.
- Tratamento em banda base: Onde se efetua o acondicionamento do sinal em banda base;
- Tratamento em radiofrequência: Aplicação da portadora de 5,8 GHZ ao sinal, amplificação e transmissão.

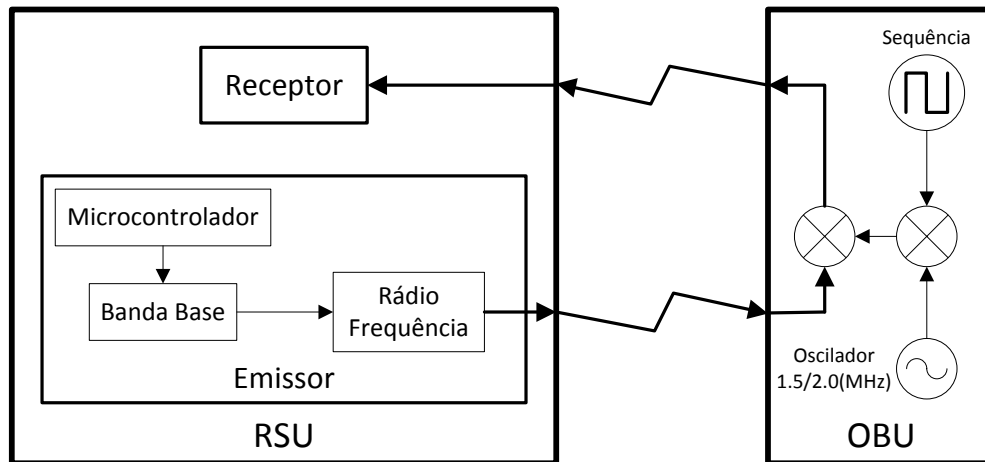


Figura 2.3: Arquitetura de um sistema DSRC (pormenorizada)

No caso particular deste projeto pretende-se juntar os blocos correspondentes ao microcontrolador e ao tratamento do sinal em banda base numa FPGA. Contudo, há que ter em conta que apenas se considera as camadas e configurações que podem ser implementadas a nível do *hardware* (FPGA), isto é, a camada física e a camada MAC. As camadas superiores seriam sempre implementadas num microcontrolador por tal ser mais simples e eficaz, porém essas camadas não fazem parte dos objetivos traçados para o trabalho que se irá realizar. Logo não haverá a necessidade de se utilizar um microcontrolador no trabalho.

O desenho da OBU não é objeto de estudo neste projeto, mas importa referir que este é projetado de maneira a que o seu custo e dimensões sejam os menores possíveis. Além disso alguns pormenores acerca do seu funcionamento resultantes da observação da figura 2.3 podem ser dados. À semelhança da RSU também esta é constituída pelos três blocos acima descritos. Porém ao invés de na figura 2.3 se ilustrar uma OBU constituído por os três blocos optou-se por ilustrar as partes importantes de uma OBU aquando de uma transmissão, de maneira a salientar a importância de se fazer o envio da portadora por parte do emissor da RSU. Isto é, uma das características da OBU é não possuir um oscilador interno de 5,8 GHz, logo para esta conseguir enviar corretamente um sinal terá de aproveitar a portadora enviada pela RSU, funcionando esta como o oscilador de 5,8 GHz. Assim apenas se necessitará de um oscilador interno de 1,5 MHz ou 2,0 MHz, necessidade essa que será explicada mais à frente.

2.2.2 Protocolo

Como já referido foi feito um esforço por parte da comissão europeia para permitir a interoperabilidade de vários sistemas EFC em diferentes países, obrigando a utilização de uma ou mais das tecnologias acima já enunciadas. No caso do DSRC foram criadas várias normas de maneira a definir cada uma das camadas OSI do protocolo. O DSRC, estando incluído num sistema *embedded*, é definido por três camadas do modelo OSI: a camada física, a camada de ligação de dados e a camada de aplicação. De seguida é feita uma breve caracterização de cada uma delas e são ainda referenciadas duas outras normas associadas a este protocolo, a norma EN 13372 e a EN 300 674.

A norma EN 12253 define a camada física (layer 1) do protocolo DSRC a 5,8 GHz. Esta camada tem como objetivo caracterizar variados parâmetros importantes para a transmissão

do sinal, tais como, polarização, máscaras espectrais, níveis de potência, ritmo de transmissão, modulação e codificação.

Já a camada de ligação de dados (layer 2) é caracterizada pela norma EN 12795. Esta camada engloba duas sub-camadas. De um lado tem-se a camada de controlo de acesso médio (MAC) que coordena o acesso à camada física, o endereçamento e verifica a ocorrência de erros de transmissão. Enquanto que do outro lado aparece a camada de ligação de controlo lógico (LLC) responsável por o estabelecimento das ligações entre a RSU e a OBU.

No que à camada de aplicação (layer 7) diz respeito a norma que a define é denominada por EN 12834. Esta norma não foi objeto de estudo do trabalho desenvolvido é contudo enriquecedor referi-la. Nesse sentido tem-se a argumentar que este padrão fornece as ferramentas necessárias para as comunicações baseadas em DSRC. Estas ferramentas consistem em elementos que podem ser usados em processos de várias aplicações, de maneira a permitir o início das comunicações, a transferência de dados e operações remotas. Possibilitam ainda a ocorrência de múltiplas aplicações simultâneas, intercalando umas com as outras.

Por fim temos as normas EN 13372 e a EN 300 674. A primeira define o perfil das comunicações RTTT baseadas no protocolo DSRC, funcionando como um complemento às normas acima referidas. Este padrão apresenta um certo número de conjuntos de parâmetros que podem ser usados nas comunicações entre uma RSU e uma OBU e associa um identificador único a cada conjunto. Este identificador é usado em negociações e nos processos de inicialização entre a unidade fixa e a unidade móvel. Já a segunda norma define quais os testes e medidas que deverão ser feitos para garantir que os equipamentos criados estão de acordo com as regras impostas pelo ETSI.

Como o projeto se foca mais na camada física do protocolo DSRC, englobando também alguns aspetos da camada de ligação de dados, apenas se fará uma referência mais pormenorizada às normas que definem estas duas camadas.

2.2.3 Camada física

Taxa transmissão e preâmbulo

No downlink (da RSU para a OBU) a taxa de transmissão é de 500 kbits, ou seja, um bit a cada $2 \mu s$.

Há necessidade de no início de cada comunicação entre a RSU e a OBU enviar um preâmbulo que terá um comprimento de 16 bits e uma duração de $32 \mu s$.

Modulação

A modulação a ser aplicada ao sinal no emissor é do tipo ASK. Neste género de modulação é aplicado um valor de amplitude ao nível alto do sinal e outro, diferente, ao nível baixo (modulação na amplitude). A relação entre estes dois níveis é-nos dada pelo índice de modulação (m). Esse índice é obtido a partir da formula 2.1, onde v_{max} correspondente ao nível mais alto de tensão envolvente e v_{min} ao nível mais baixo de tensão envolvente.

$$m = \frac{v_{max} - v_{min}}{v_{max} + v_{min}} \quad (2.1)$$

O índice de modulação deverá situar-se entre os '50%' e os '90%'.

Codificação

De forma a conferir ao sinal certas características úteis do ponto de vista do recetor (remoção da componente contínua (DC) do sinal e existência de componente espectral à frequência do relógio de transmissão) é ainda aplicada uma codificação em banda base. Neste caso é utilizada a *FM0* que consiste numa variação de nível sempre que há um novo bit a ser enviado. Se esse bit for 1 o nível mantém-se ao longo do envio desse bit, se for um 0 irá ocorrer uma transição a meio do bit. Na figura 2.4 ilustra-se a comparação entre um sinal com e sem codificação *FM0*.

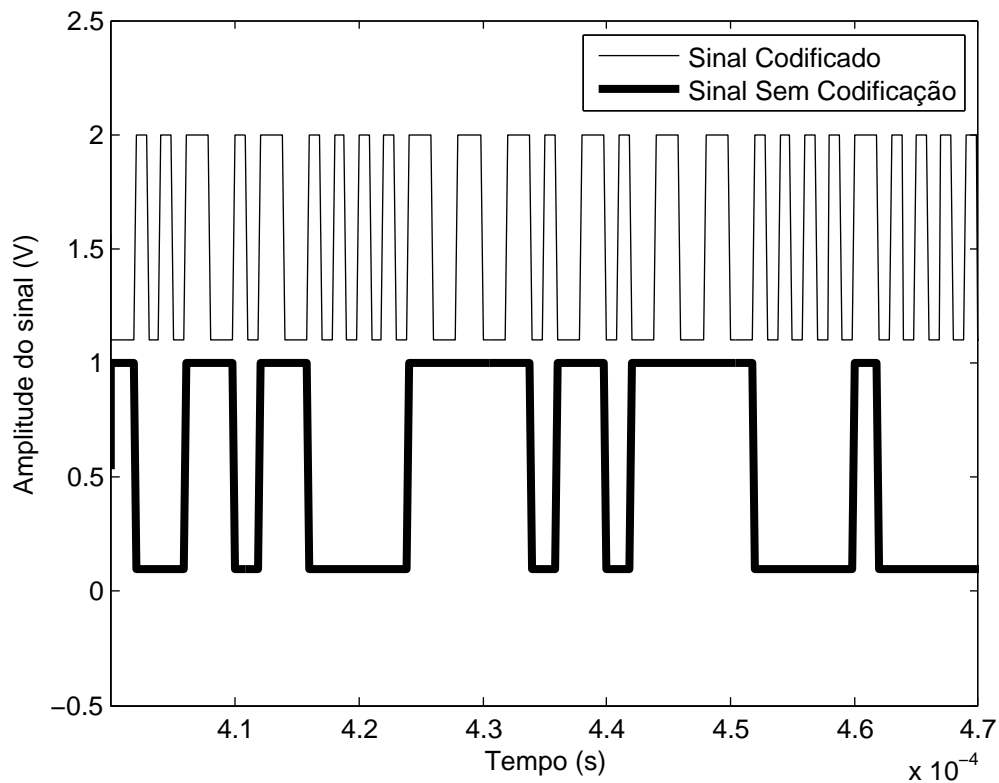


Figura 2.4: Exemplo de uma codificação FM0

A figura 2.4 ilustra como é feita a codificação. A escala do tempo está dividida em fracções de 10 μ s, ou seja, tem-se cinco bits de informação por cada divisão de escala.

Máscaras espectrais

Em relação à potência transmitida no sinal, esta não deverá ultrapassar os +33 dBm de PIRE.

As regiões fora da banda de passagem deverão respeitar as máscaras espectrais definidas pela norma. Existem três classes possíveis, para as máscaras espectrais de transmissão da RSU. As classes e respetivas máscaras espectrais estão representadas nas tabelas 2.1, onde a classe C é a mais restritiva e a A a menos.

Posição	Frequências	Desmodulada		Modulada		L.B. Equivalente
		Classes A,B,C	Classe A	Classe B	Classe C	
Cocanal	1.0Mhz	-27dBm	–	–	–	62.5Khz
Cocanal	1.5Mhz	-27dBm	-7dBm	-17dBm	-27dBm	500Khz
Cocanal	2.0Mhz	-27dBm	-27Bm	-27dBm	-27dBm	500Khz
Canais adjcs	3.0Mhz	-47dBm	-30dBm	-37dBm	-47dBm	500Khz
	3.5Mhz					
	6.5Mhz					
	7.0Mhz					
Canais adjcs	4.0Mhz	-47dBm	-30dBm	-37dBm	-47dBm	62.5Khz
	6.0Mhz					

Tabela 2.1: Máscaras espectrais e suas diferentes classes

No âmbito deste projeto de forma a criar um emissor que seja ótimo em variados sistemas vamos optar por seguir a classe mais restritiva, ou seja, a classe C.

De maneira a facilitar e a melhor clarificar a leitura e interpretação das máscaras espectrais acima descritas, apresenta-se sob a forma de figuras essas mesmas máscaras (figuras 2.5 e 2.6). A figura 2.5 ilustra o caso LDR, ao passo que a figura 2.6 apresenta o caso relativo ao MDR.

As designações LDR e MDR estão relacionadas com as taxas de transmissão aplicadas. No caso do presente trabalho estar-se-á no MDR, contudo é bom grado respeitar os limites impostos por ambos os casos.

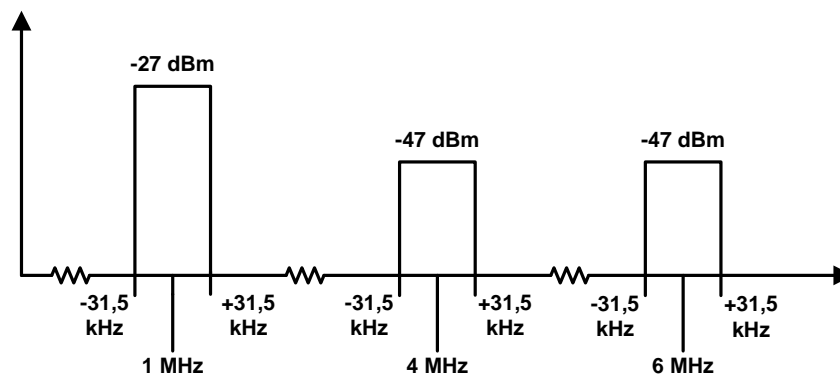


Figura 2.5: Máscaras espectrais (LDR)

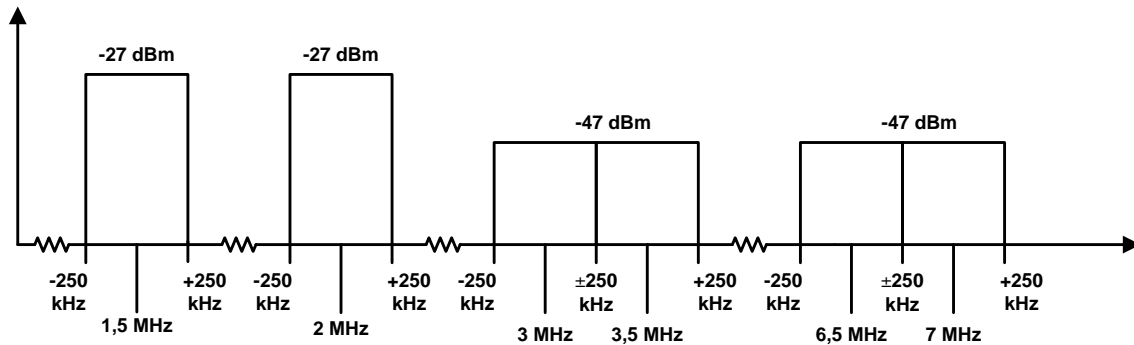


Figura 2.6: Máscaras espectrais (MDR)

Em relação ao resto do espectro são ainda especificados limites para as emissões indesejadas (Tabela 2.2).

Bandas de frequência	Limites (dBm)	Largura de Banda (kHz)
47Mhz até 74Mhz	-54	100
87.5Mhz até 118Mhz		
174Mhz até 230Mhz		
470Mhz até 862Mhz		
Frequências >30Mhz e ≤1Ghz	-36	100
Frequências >1Ghz e <26Ghz	-30	1000

Tabela 2.2: Máscaras espectrais para todo espectro

Canais de comunicação

Existem formas de realizar um eficiente e fidedigno processamento num sistema *multilane* (várias comunicações simultâneas), sem impor regras como redução de velocidade ou barreiras físicas. Estas implementações destinam-se assim a reduzir a interferência entre veículos, ou seja, isolar as comunicações (*uplink* e *downlink*) umas das outras.

O protocolo DSRC disponibiliza quatro frequências para a portadora, espaçadas de 5 MHz em torno dos 5,8 GHz, para a realização das comunicações de *downlink*. Estas frequências encontram-se escalonadas em:

- Canal 1 de *downlink*: 5,7975 GHz;
- Canal 2 de *downlink*: 5,8025 GHz;
- Canal 3 de *downlink*: 5,8075 GHz;
- Canal 4 de *downlink*: 5,8125 GHz.

Os canais 1 e 2 são especificados na norma correspondente à camada física, enquanto que os canais 3 e 4 são de utilização facultativa.

No caso do *uplink* outra técnica é usada. Para cada uma das frequências de portadora de *downlink* usadas estão disponíveis duas frequências de *uplink*, afastadas 1,5 MHz ou 2

MHz em relação à frequência da portadora de *downlink*. Ou seja, a portadora enviada pela RSU é aproveitada pela OBU, que por sua vez a remodulada com uma sub-portadora gerada localmente. A figura 2.7 ilustra duas portadoras de *downlink* e as respectivas sub-portadoras afastadas 1,5 MHz ou 2 MHz.

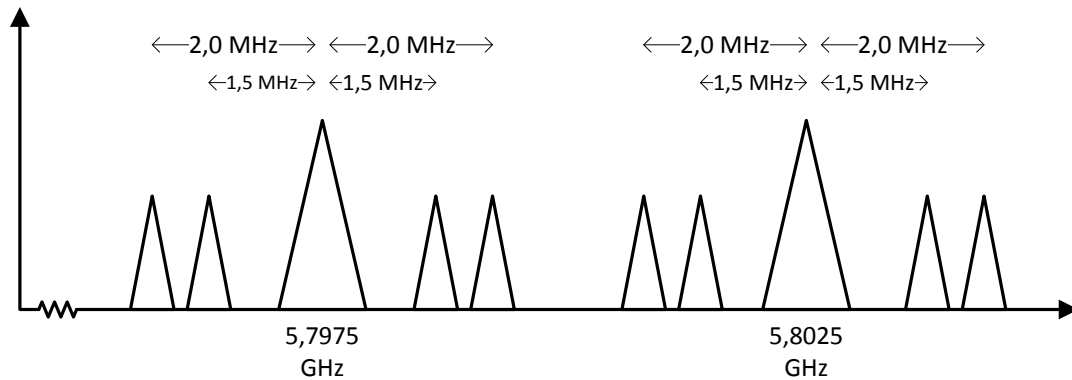


Figura 2.7: Portadoras e sub-portadoras de uma comunicação DSRC

Consoante a portadora e sub-portadora utilizadas diferentes denominações são dadas às outras sub-portadoras. Existem dois tipos de denominações: o cocanal e o canal adjacente.

O cocanal corresponde à outra sub-portadora associada à mesma portadora que se está a usar. Ou seja, no caso de numa dada comunicação se estar a usar a portadora 5,7975 GHz e a sub-portadora de 1,5 MHz o cocanal corresponderá à sub-portadora de 2,0 MHz.

Por sua vez denomina-se de canal adjacente as sub-portadoras associadas a uma portadora que não seja a utilizada. Tomando o exemplo anterior, um canal adjacente em relação a esse exemplo será a sub-portadora de 1,5 MHz associada à portadora de 5,8025 MHz.

2.2.4 Camada de ligação de dados

No que à camada MAC diz respeito deve-se ter em conta também alguns aspetos por ela definidos.

Flags de início e fim de trama

No envio de uma mensagem é necessário a introdução de *flags* para sinalizar o início e o fim da mesma. A figura 2.8 representa o formato da *flag* de início e fim de trama utilizada.

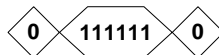


Figura 2.8: Exemplo de uma *flag* de sinalização

Bit Stuffing

Deve-se prevenir a ocorrência de uma sequência idêntica às *flags* no interior da trama de informação. É por isso necessária a introdução de um bit de *stuffing*, ou seja, um bit a 0 que não representa informação válida. Este bit de “lixo” deve ser introduzido sempre que dentro

de uma trama apareçam cinco bit's a 1 consecutivos. Isto porque as *flags* de inicialização apresentam seis bits a 1, como é visível na figura 2.8.

2.3 Sistemas *software defined radio*

Os primeiros sistemas SDR apareceram em meados dos anos 80. O *SpeakEasy* foi um dos primeiros equipamentos criados com base nesta tecnologia. Este dispositivo foi criado pela *Hazeltine* e pela *Motorola* para a base da força aérea, Rome, de Nova Iorque. O *SpeakEasy* foi desenvolvido para fins militares, ou seja, para permitir comunicações militares habilidosas e permitir também a interoperabilidade entre diferentes protocolos de interfaces aéreas de diferentes forças armadas. Todavia, apesar do *SpeakEasy* ser um sistema completamente desenvolvido em SDR, é necessário referir que outros equipamentos mais simples, rudimentares e de maiores dimensões precederam este.

Já entre 1997 e 1998 surgiu outro projeto denominado por sistema de rádio táctico conjunto (JTRS). Este projeto foi criado em conjunto pela organização do tratado do atlântico norte (NATO) e os Estados Unidos e tinha por objetivo produzir rádios flexíveis e interoperacionais entre eles recorrendo ao uso de um *software* de arquitetura de comunicações (SCA) aberto. O JTRS apesar de ser um projeto de âmbito militar encontra-se disponível para fabricantes de sistemas de rádios, podendo ser englobado por estes nos seus sistemas se tal for desejo do fabricante.

Desde então os sistemas SDR têm vindo a desenvolver-se e começam a ser cada vez mais uma realidade nos tempos de hoje. Contudo não existe ainda uma definição padrão para o termo *software defined radio*, assume-se que um sistema de rádio é definido por *software* se se puder fazer o tratamento do sinal através de *software*, isto é, se for possível fazer o acondicionamento do sinal em banda base e/ou fazer algum tratamento de radiofrequência sem ser necessário adicionar *hardware* específico para esse fim, apenas recorrendo a *software*.

O objetivo destes sistemas é então a implementação em domínio digital das operações necessárias ao tratamento do sinal, ao invés de se o fazer em domínio analógico. Neste sentido um sistema SDR refere-se a um rádio configurável/reprogramável, sendo por isso um rádio muito flexível que pode ser modificado para desempenhar diferentes funções.

Contudo o tratamento ao nível das radiofrequências é ainda efetuado por dispositivos analógicos, existindo assim a necessidade de se acrescentar um *front-end* para esse fim.

Do uso de equipamentos baseados em *software defined radio* advêm várias vantagens:

1. O uso desta tecnologia permite uma economia de tempo no desenvolvimento de novos produtos pois não é necessário criar um novo projeto, com novos componentes analógicos sempre que se pretende fazer alguma alteração ao projeto inicial, para além de que o problema de ser altamente improvável a construção de dois componentes analógicos idênticos é eliminado com o uso de sistemas SDR.
2. O mesmo dispositivo pode operar de acordo com vários protocolos, pois basta descarregar o correto *software* na memória do sistema e este passa a trabalhar de acordo com o especificado, sem ser preciso substituir ou retirar o aparelho do local onde se encontra inserido.
3. A existência de componentes digitais de elevada capacidade e velocidade possibilita a implementação das técnicas utilizadas na transmissão e recepção de dados de uma forma simples eficaz e eliminam a necessidade de utilização de muitos componentes discretos.

2.3.1 Utilização de FPGAs em SDR

O elemento que se utilizará, no âmbito deste projeto, para efetuar o tratamento digital do sinal será uma FPGA. Estes dispositivos são circuitos digitais que podem ser programados de acordo com as necessidades de cada utilizador. O que permite esta adaptabilidade é a sua estrutura singular. Ou seja, uma FPGA é constituída por vários blocos lógicos, que por sua vez contêm no seu interior blocos combinatórios simples, e *flip-flops*, que podem ser interligados de maneira a criar circuitos mais complexos. Os próprios blocos lógicos estão inseridos numa malha que permite a sua interligação. As FPGAs de hoje em dia apresentam também blocos de memória dedicados permitindo uma optimização dos circuitos implementados.

Algumas das vantagens das FPGAs são:

- Grande nível de paralelismo: Permite que haja vários processos a ser executados no mesmo instante, o que representa uma grande vantagem na implementação de filtros e, por isso, na implementação de sistemas *software defined radio*. Esta característica representa também um aumento de desempenho em comparação com um DSP com a mesma velocidade de processamento;
- Implementação de operações específicas: Isto é, no caso de se querer implementar uma operação que não seja facilmente mapeável a sua execução será mais rápida e eficiente numa FPGA do que num DSP;
- As empresas distribuidoras da tecnologia oferecem plataformas para o desenho e desenvolvimento dos projetos que irão ser implementados (ferramentas de projeto electrónico (EDAs)), documentação necessária e completa e mecanismos de apoio ao cliente;
- Reconfiguração e adaptabilidade: Oferecem a mais valia de se poder modificar e ajustar o projeto inicial a qualquer altura, mesmo quando este já se encontra instalado. Sendo que em FPGAs mais recentes é mesmo possível fazer reconfigurações dinâmicas em equipamentos já ativos, isto é, há a possibilidade de alterar partes do projeto implementado no interior da FPGA sem ser necessário para isso o carregamento do novo projeto na sua totalidade;
- Redução de custos sobre direitos: Para a configuração da FPGA é gerado um ficheiro de configuração específico, sendo que o projeto em código VHDL é mantido do lado de quem desenvolve mantendo-se assim confidencial. Não há necessidade por isso da compra dos direitos sobre esse código por parte de quem irá comprar o equipamento, reduzindo-se assim o custo associado.

2.3.2 Arquitetura do emissor com base em SDR

A arquitetura do emissor a ser implementado irá ser a representada na figura 2.9. Esta é uma arquitetura já com base na tecnologia SDR, pelo que se faz uso de uma FPGA para se efetuar a modulação tanto da parte do *hardware* necessário para o projeto do emissor, como também de alguns aspetos relacionados com o *software*.

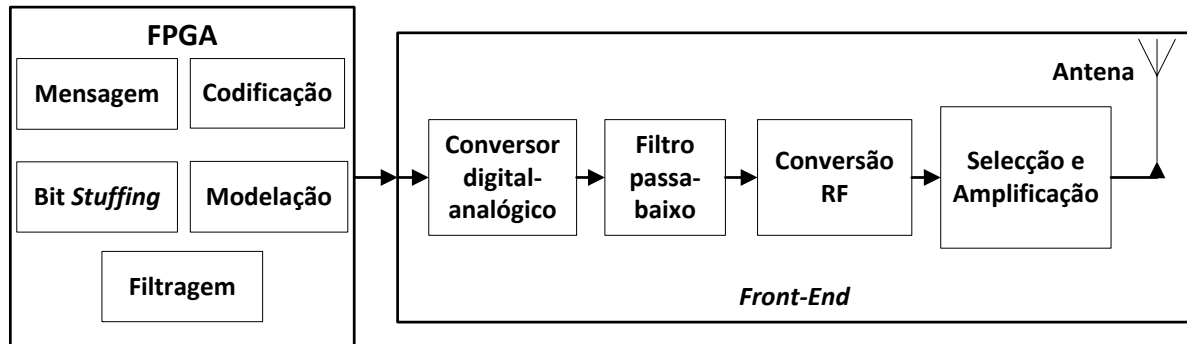


Figura 2.9: Emissor baseado na tecnologia SDR

Nesta implementação juntar-se-á na FPGA algumas das funções usualmente desempenhadas pelo microcontrolador e o acondicionamento feito a nível analógico do sinal em banda base.

A incorporação de algumas das funções do microcontrolador na FPGA irá providenciar ao sistema um maior desempenho sem com isso ser necessário um melhor processador. Este aumento de desempenho confere também uma maior fiabilidade ao sistema, tendo em conta que a RSU é para ser utilizada em autoestradas (trânsito *free-flow* e altas velocidades). Além disso, como algumas das especificações que se têm de cumprir na camada MAC irão ser englobadas na camada física e na FPGA, irá-se ter uma camada MAC mais “leve” e menos complexa.

De referir também que toda a parte de tratamento do sinal em banda base será feita com base na tecnologia SDR, como é o caso da codificação, modulação do sinal e também filtragem do sinal, deixando apenas para o lado analógico alguma filtragem em banda base (a sua necessidade é explicada mais à frente no documento) e a parte referente à radiofrequência. Ou seja o front-end correspondente à parte analógica terá o formato mostrado na figura 2.9.

2.3.3 Conversor digital-analógico

Como grande parte do tratamento do sinal é feito através de *software defined radio*, é necessário assegurar que, na passagem de digital para analógico, esse trabalho é prejudicado o menos possível pelo conversor digital-analógico (DAC).

Grande parte do problema está relacionado com o números de bits limitado que é utilizado pelas DACs para efetuar a conversão, além das não linearidades intrínsecas a estas. Estas limitações podem, muitas vezes, levar a que a DAC seja o elemento responsável por limitar o desempenho de todo o sistema e impedi-lo de alcançar o seu objetivo.

Cria-se assim a obrigação de se utilizar DACs que não comprometam o desempenho global do sistema. Nesse sentido há já um esforço por parte dos distribuidores em produzir DACs ideais para sistemas *software defined radio*, ou seja, conversores que operem a altas frequências, e que permitam que a parte digital do sistema se aproxime mais da antena. Há assim uma maior redução da parte analógica envolvida no sistema.

Relação entre o sinal e o ruído de quantificação

Neste trabalho apenas se terá em conta a relação entre o sinal e o ruído de quantização (SQNR). Esta relação é útil para analisar a quantidade de distorção introduzida pela quantificação do sinal.

A SQNR é definida como a relação entre a potência do sinal (P_x) e a potência do ruído (N_Q).

$$SQNR = \frac{P_x}{N_Q} \quad (2.2)$$

Assumindo como entrada no conversor um sinal uniformemente distribuído, a potência média do sinal é dada pela sua variância, isto é

$$P_x = \alpha_x^2 = E[x^2] = \frac{P_{FS}^2}{12} \quad (2.3)$$

Usando a relação dada pela equação 2.3, a SQNR para um sinal de entrada uniformemente distribuído aplicado a um quantizador uniforme com B bits de quantização pode ser calculada por

$$SQNR = \frac{\frac{P_{FS}^2}{12}}{\frac{P_{FS}^2}{12(2^{2B})}} = 2^{2B} = 10 * \log_{10}(2^{2B})dB \quad (2.4)$$

2.3.4 Filtros em SDR

Neste projeto os filtros desempenham um importante papel, isto porque, como já discutido na norma EN 12253 (camada física), o sinal tem que obedecer a determinados limites de potência. Logo com vista a cumprir este objetivo o uso de filtros para limitar a potência transmitida é imperativo, sendo por isso conveniente fazer referência a tais componentes. Igualmente o uso habitual de um conversor digital-analógico em SDR introduz a necessidade de se utilizar um filtro para a reconstrução do sinal à sua saída.

Faz-se agora uma pequena abordagem a estes componentes, referindo as suas características e diferenças entre uma abordagem a nível analógico e a nível digital.

Os filtros electrónicos permitem seleccionar quais os sinais que deverão passar e quais deverão ser descartados. Mais concretamente os filtros permitem apenas que determinadas frequências do sinal passem da sua entrada para a sua saída com pouco ou nenhuma redução do nível do sinal [3].

Num filtro a gama de frequências que deverão passar pelo filtro sofrendo pouca ou nenhuma redução do nível do sinal fazem parte da banda de passagem. A frequência de corte do filtro corresponde ao ponto em que o nível do sinal, após a banda de passagem, tem uma redução de 3 dB. A banda de frequências que se pretende descartar compõe a zona de corte do filtro. À zona que se encontra entre estas duas frequências dá-se o nome de zona de transição. Nesta zona há uma grande variação na atenuação do sinal.

Os filtros podem apresentar diferentes respostas em frequência:

- Filtros passa-baixo: Permitem a passagem das baixas frequências, isto é, deixam passar todas as frequências desde a componente contínua (DC) até à frequência de corte.

- Filtros passa-alto: São o oposto dos filtros passa-baixo, deixam passar todas as frequências acima da frequência de corte.
- Filtros passa-banda: Deixam passar as frequência que se encontram intercaladas entre as frequências de corte inferior e superior.
- Filtros rejeita-banda: Impedem a passagem das frequências que se encontram intercaladas entre as frequências de corte inferior e superior, deixando passar todas as outras.

Os filtros podem ser implementados a nível digital ou a nível analógico.

Filtros Analógicos

Os filtros analógicos podem ser do tipo ativo ou do tipo passivo. Os passivos apenas fazem uso de resistências, condensadores e bobinas. Estes filtros são comumente utilizados para a implementação de filtros que operam a altas frequências e/ou necessitam de uma grande gama dinâmica (diferença entre o ruído de fundo e a amplitude máxima do sinal). Têm ainda a grande vantagem de não exigirem alimentação.

Os filtros analógicos do tipo ativo usam amplificadores operacionais como o elemento ativo do circuito. Os amplificadores são combinados com resistências e condensadores originando um filtro com a resposta em frequência apropriada, eliminando, por vezes, o uso de bobinas. Estes filtros encontram-se limitados ao nível das frequências de operação permitidas, devido às limitações de ganho e largura de banda dos amplificadores, sendo utilizados em implementações abaixo das dezenas de megahertz (MHz). Com o uso de filtros ativos tem-se a vantagem de se conseguir filtros de menores dimensões em comparação com os filtros passivos. Contudo devido ao uso de amplificadores há o inconveniente de se acrescentar ruído ao sinal e de se limitar a sua amplitude à saída devido tanto à alimentação do amplificador, como ao *slew-rate* do mesmo. O uso do amplificador pode acrescentar ainda alguma distorção no sinal. Os filtros analógicos do tipo ativo são por isso mais comumente usados em implementações pouco exigentes, onde as mudanças de amplitude dos sinais sejam graduais com a frequência.

Filtros Digitais

Os filtros digitais estão a tornar-se cada vez mais comuns nos dias de hoje, substituindo já os filtros analógicos em muitos sistemas.

As vantagens de se utilizar filtros digitais passam pelo facto destes apresentarem uma resposta em frequência reproduzível, não serem sensíveis à temperatura e serem programáveis. Em casos em que o processamento do sinal é efetuado a nível digital a utilização destes filtros é a escolha mais óbvia, pois não haverá a necessidade de converter o sinal para analógico. Os filtros digitais conseguem conciliar uma zona de corte estreita em frequência com uma fase linear no domínio do tempo.

Contudo os filtros digitais exigem uma amostragem do sinal se este for analógico, requerem o uso de uma fonte de alimentação e apresentam limites na gama de frequências a que podem operar (está dependente da resolução necessária para a gama dinâmica do sinal digital).

2.3.5 Outros Projetos baseados em SDR

Outros projetos que fazem uso de sistemas SDR são os de âmbito militar. Um deles já referido na secção referente à história do SDR é o JTRS. O *paper* [4] fala um pouco dos

sistemas SDR usados e como certas técnicas podem ser implementadas fazendo uso destes sistemas.

Já o *paper* [5] centra-se nas especificações da arquitetura do *software* de um sistema SDR, mais concretamente na necessidade de se definir um conjunto de APIs e escaloná-las em classes tendo em conta, por exemplo, o custo e a capacidade.

Capítulo 3

Modelação e simulação do emissor

3.1 Introdução

Neste capítulo é feita a primeira abordagem ao projeto do emissor, recorrendo a simulações efetuadas em ambiente *Matlab*. Estas simulações têm como objetivo não só estudar o comportamento das máscaras espectrais do sinal emitido pelo emissor, mas também ajudar na correta escolha do *front-end* que se terá de projetar para se fazer a conversão digital-analógico. Como já referido, o conversor digital-analógico (DAC) representa um elemento de elevada importância em todo o sistema, contribuindo para o seu bom desempenho.

3.2 Modelação

É feita agora a apresentação do emissor completo a ser implementado e refere-se quais os blocos que efetivamente vão ser modelados em *Matlab*. Aborda-se também qual será a estrutura do modelo simulado.

3.2.1 Arquitetura do sistema

A figura 3.1 ilustra o emissor completo, ou seja, apresenta as várias fases pelas quais o sinal irá passar até ser emitido para o meio. De maneira a melhor situar o leitor, convém referir que as simulações feitas referem-se apenas aos primeiros dois blocos ilustrados na figura (FPGA e DAC). Irá somente estudar-se qual a influência que esses blocos têm no comportamento do sinal, mais concretamente, no seu espectro.

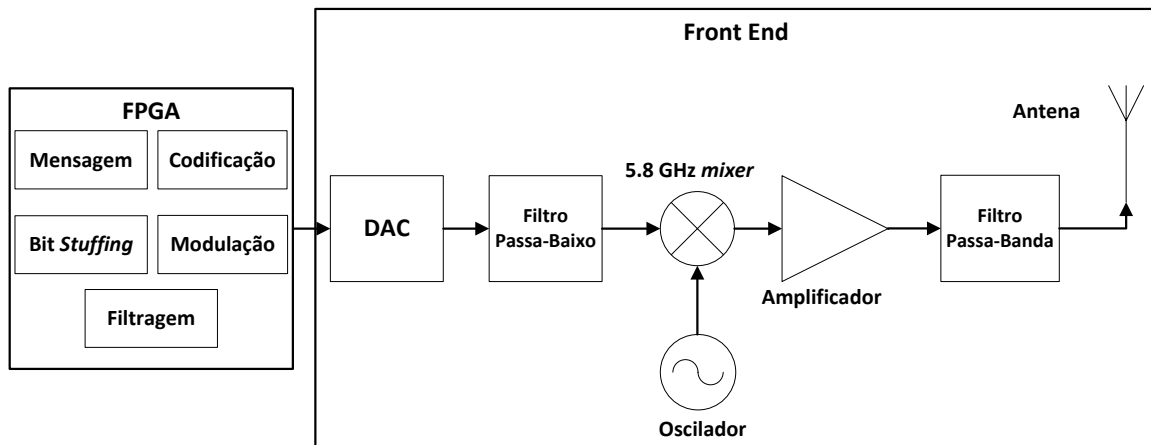


Figura 3.1: Diagrama geral de blocos do emissor

De referir que o oscilador, o *Mixer*, o amplificador, o filtro passa-banda e a antena não serão alvo de atenção ao longo da elaboração do trabalho. Consistem em blocos já existentes no laboratório e que poderão ser adicionados aos restantes de forma a testar o sistema completo quando tal for necessário. Já o filtro analógico, apesar de se fazer uso de um já existente, desempenhará um papel importante como irá ficar claro com as simulações efetuadas.

3.2.2 Modelo implementado no *Matlab*

Na figura 3.2 está ilustrado o sistema que foi efetivamente modelado em *Matlab*.

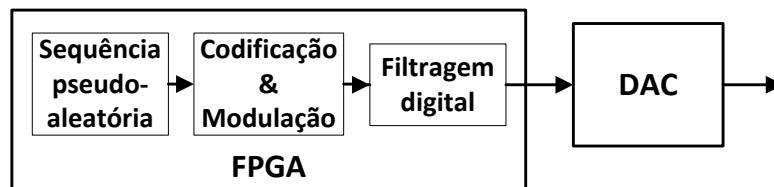


Figura 3.2: Sistema modelado

Os blocos simulados serão então:

FPGA

Este será o bloco sobre o qual recairá maior parte da atenção durante a elaboração do trabalho. A FPGA será responsável pela maior parte do tratamento aplicado ao sinal a enviar (modulação, codificação, *bitstuffing*), contudo o mais importante é a filtragem que será efetuada a nível digital. O objetivo principal será, por isso, avaliar qual a melhor solução a tomar de maneira a se obter um bom desempenho na filtragem.

A discussão já feita no final do capítulo 2 entre filtros digitais e analógicos, mostrou que a escolha de uma implementação baseada em filtros digitais é preferível quando comparada com uma baseada em filtros analógicos. Pelo que haverá filtragem a nível digital, ou seja, no interior da FPGA. A escolha de filtros digitais é vantajosa pois no caso específico deste

trabalho há já necessidade de se recorrer a uma unidade alimentada, (FPGA), não haverá necessidade de amostrar o sinal pois ele já se encontra em formato digital, as frequências utilizadas não serão elevadas e está-se a fazer um projeto com base em SDR.

DAC

Irão ser feitas simulações considerando DACs de 12, 10 e 8 bits. É importante fazer-se a simulação da DAC devido ao ruído de quantificação que poderá ser introduzido e por ser um equipamento que depois de montado em PCB não poderá ser alterado.

3.2.3 Estrutura do modelo do *Matlab*

A figura 3.3 ilustra o modelo que foi implementado no *Matlab*. São especificados os blocos criados e a ordem pela qual eles estão encadeados.

No anexo A encontra-se descrito o código *Matlab* utilizado.

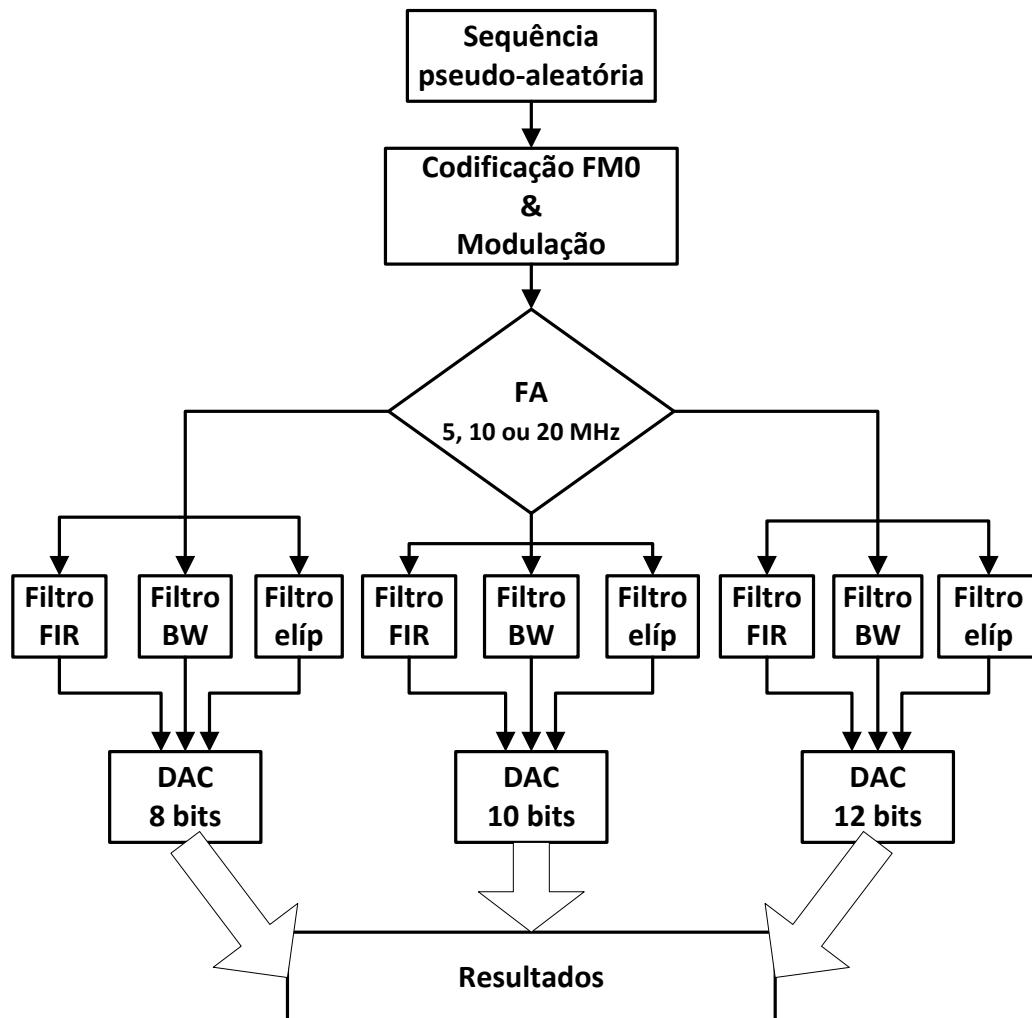


Figura 3.3: Diagrama de fluxo do modelo do *Matlab*

Faz-se de seguida a descrição do modelo e blocos acima expostos, de maneira a que o leitor melhor entenda o sistema modelado no *Matlab*.

Criação da sequência

A sequência elaborada é simples. Criou-se um vetor aleatório de bits a 0 e 1, aos quais se aplicou uma codificação FM0. Posteriormente modulou-se o sinal, ou seja, alterou-se o valor dos bits por forma a se ter um índice de modulação da ordem dos 90%.

Filtragem

Irão ser modelados em *Matlab* dois tipos de filtro digitais, que apresentam desempenhos e implementações diferentes. São eles os filtros de resposta impulsional finita (FIR) e os de resposta impulsional infinita (IIR). O funcionamento de um filtro FIR consiste basicamente na recepção de amostras, processamento das mesmas e posterior apresentação destas à saída. Estes filtros apresentam as vantagens de terem uma resposta com fase linear e estável. São também insensíveis a efeitos de quantificação e além disso não possuem o problema de poderem oscilar. Uma última vantagem é o facto de haver *cores* pré-fabricados que tornam a utilização destes filtros muito mais simples.

Já a implementação de um filtro IIR é recursiva, ou seja, a saída é realimentada na entrada, sendo que é esta estrutura que leva a que o filtro possa oscilar. No entanto estes filtros têm as vantagens de ter uma melhor atenuação e necessitarem de menos coeficientes logo menos memória. Para além disso apresentam menor latência e a sua estrutura é mais próxima dos modelos analógicos.

Em relação aos filtros IIR vão ser modelados dois tipos: filtros butterworth e filtros elípticos.

- Filtros elípticos

Os filtros elípticos apresentam ripple tanto na zona de passagem como na de corte e são, de entre todos os filtros, os que conseguem atingir os objetivos, para os quais foram desenhados, com recurso ao menor número de coeficientes. Apresentam ainda uma curta zona de transição.

- Filtros Butterworth

Os filtros Butterworth não têm ripple nas zonas de passagem e corte e apresentam a melhor aproximação à resposta ideal de um filtro passa-baixo analógico. Apresentam contudo uma zona de transição mais suave, demorando mais a atingir a atenuação pretendida.

O objetivo desta filtragem é o cumprimento das máscaras espectrais referidas na norma EN 12253 relativa à camada física do DSRC.

A caracterização e implementação dos filtros será abordada mais à frente.

Conversor digital-analógico

Este bloco tem como objetivo modelar o funcionamento de uma DAC de 12, 10 e 8 bits.

O seu funcionamento consistirá basicamente em quantificar o sinal à saída dos filtros em 4096, 1024 e 256 níveis diferentes, simulando assim o funcionamento de uma DAC o que

permitirá observar qual a influência do ruído de quantificação no comprimento das máscaras espectrais.

Frequência de amostragem

O modelo apresentado na figura 3.3 é simulado três vezes, ou seja, são feitos os mesmos passos para o caso de uma frequência de amostragem de 5, 10 e 20 MHz. O objetivo será observar quais as diferenças introduzidas pela variação da frequência de amostragem.

Resultados

Os resultados consistem no cálculo da potência em cada uma das bandas definidas pela norma EN 12253 (tabela 3.1). Demonstra-se agora como serão calculados os valores pretendidos.

A potência média de um sinal no tempo, considerando o intervalo $[-u, u]$, é dada pela equação 3.1.

$$P_x = \frac{1}{2u} \sum_{i=-u}^u x(i)^2, \quad (3.1)$$

donde, recorrendo ao teorema de parseval, deriva uma solução idêntica para o caso da frequência:

$$P_x = \frac{1}{2u^2} \sum_{i=-u}^u X(i)^2 \quad (3.2)$$

Assim depois de se criar o sinal de teste aplicar-se-á a este a transformada de *Fourier* e obter-se-á o vetor X sobre qual serão calculadas as potências pretendidas através da equação 3.2. No anexo A é dado a conhecer o código *MatLab* usado no cálculo dos valores apresentados nas tabelas obtidas nas simulações.

3.3 Simulação

Recorrendo ao modelo discutido na secção anterior, serão agora feitas uma série de simulações, seguindo os passos abaixo enunciados.

1. Teste das máscaras espectrais:

- Análise de filtros IIR:
 - Frequência de amostragem de 20 MHz.
- Análise filtro FIR com variação da frequência de amostragem;
 - Frequência de amostragem de 20, 10 e 5 MHz;
 - Ordem de 100.
- Comparação dos gráficos temporais do sinal antes e depois de filtrado.
 - Utilização de um filtro FIR de ordem 100;
 - Utilização de um filtro IIR Butterworth;
 - Utilização de um filtro IIR elíptico.

2. Teste das máscaras espectrais com a DAC aplicada à saída do filtro:

- Gráficos representativos do impacto da quantificação;
 - Variação do tipo de filtro (FIR, Butterworth e elíptico);
 - Frequência de amostragem do 20 MHz;
 - DAC de 8, 10 e 12 bits.
- Ruído de quantificação esperado devido à aplicação da DAC.

3.3.1 Objetivos

Nas simulações efetuadas pretendeu-se averiguar dois aspetos fundamentais:

- Quais os filtros que produzem um melhor resultado, tendo em conta as máscaras espectrais que têm de ser cumpridas;
- Qual o impacto da quantificação introduzida por uma DAC, no cumprimento das máscaras espectrais.

A tabela 3.1 apresenta os mesmos valores que a tabela 2.1, mas neste caso normalizados à potência máxima (+33 dBm). Será com estes valores que se irá comparar os resultados das várias simulações, visto que a potência aí calculada também se encontra normalizada à potência máxima.

Posição	Frequências	Não modulada		Modulada		L.B. Equivalente
		Classes A,B,C	Classe A	Classe B	Classe C	
Cocanal	1.0Mhz	-60dBc	–	–	–	62.5Khz
Cocanal	1.5Mhz	-60dBc	-40dBc	-50dBc	-60dBc	500Khz
Cocanal	2.0Mhz	-60dBc	-60Bc	-60dBc	-60dBc	500Khz
Canais adjacentes	3.0Mhz	-80dBc	-63dBc	-70dBc	-80dBc	500Khz
	3.5Mhz					
	6.5Mhz					
	7.0Mhz					
Canais adjacentes	4.0Mhz 6.0Mhz	-80dBc	-63dBm	-70dBc	-80dBc	62.5Khz

Tabela 3.1: Máscaras espectrais e suas diferentes classes (valores normalizados)

3.3.2 Filtros simulados no *Matlab*

Foram implementados em *Matlab* três tipos diferentes de filtros digitais: FIR, Butterworth e elíptico.

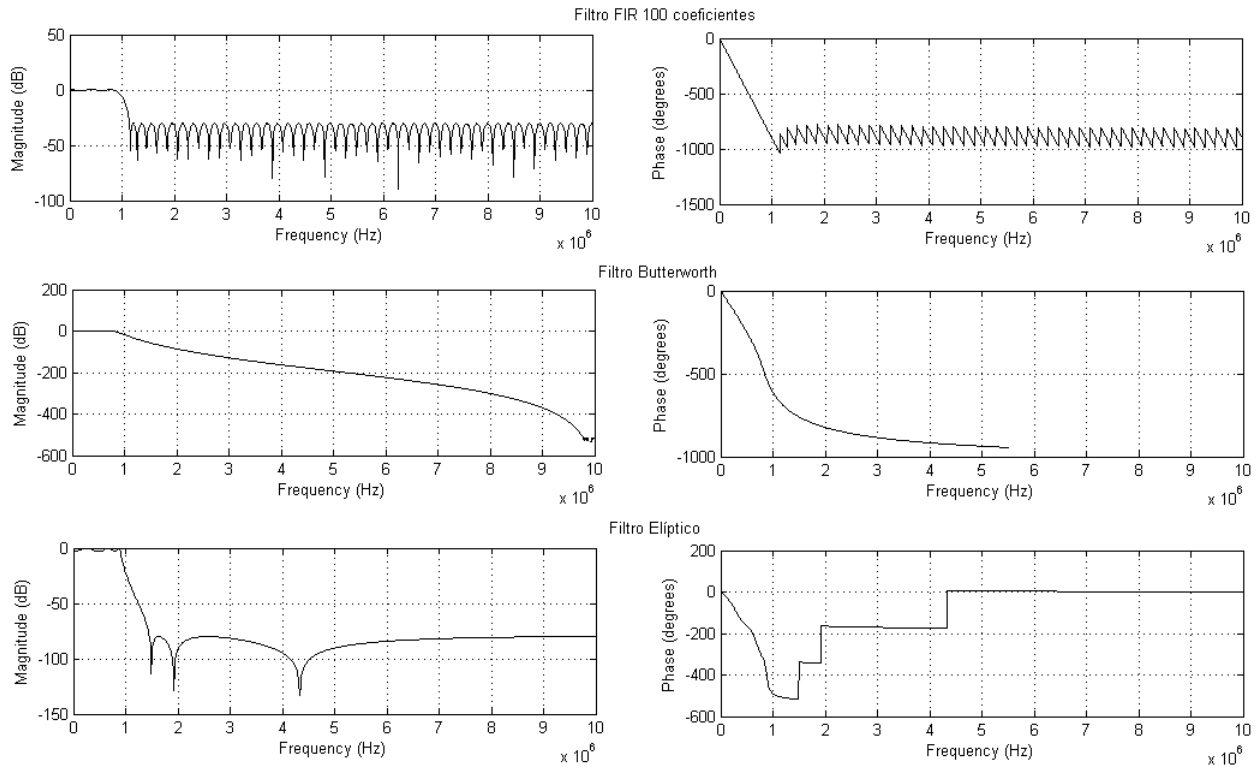


Figura 3.4: Respostas em frequência e fase dos filtros ($F_a = 20$ MHz)

A figura 3.4 apresenta as respostas de cada um dos filtros implementados para uma frequência de amostragem de 20 MHz. O mais importante a salientar da figura será a resposta em fase linear do filtro FIR e os maiores níveis de atenuação disponibilizados pelos filtros IIR.

Refere-se de seguida mais pormenorizadamente quais as especificações usadas para a criação dos vários filtros.

1. Filtros FIR

Uma das questões mais importantes na implementação de um filtro FIR é o número de coeficientes que se irá utilizar. Trata-se de uma decisão importante pois quanto mais coeficientes se usar maior terá de ser a capacidade da FPGA, já que o número de multiplicadores necessários são proporcionais ao número de coeficientes. Assim deve-se ter em conta uma abordagem que não exija a utilização de um filtro com muitos coeficientes, mas por outro lado não comprometa o desempenho do filtro (quanto mais coeficientes melhor a resposta em frequência do filtro). De maneira a se ter a melhor relação de compromisso optou-se por criar um filtro com 100 coeficientes, pois esta ordem não é elevada ao ponto de por em causa a sua implementação na FPGA e garante uma banda de passagem estreita.

Por forma a calcular-se qual a banda de passagem necessária para uma ordem de aproximadamente 100, utilizou-se a função do *Matlab* - *firpmord*. Assim basta escolher a banda de passagem e a função retorna a ordem necessária. Procedendo ao método de tentativa e erro chegou-se às seguintes especificações:

- Banda de Passagem: 0 até 0,86 MHz;

- Banda de corte: apartir dos 1,14 MHz;
- Atenuação: 30 dB;

Para as várias frequências de amostragem (5 MHz, 10 MHz e 20 MHz) manteve-se os mesmos critérios. O código usado para a criação do filtro no *Matlab* foi:

```
h1=firpm(100,[0 (0,86MHz/(fa/2)) (1,14MHz/(fa/2)) 1],[1 1 0 0])
```

2. Filtros Butterworth

Para o caso de um filtro IIR a ordem já não é problema pois geralmente é baixa. Logo recorreu-se à função existente no *Matlab* (*buttord*) e calculou-se a ordem e parâmetros do filtro Butterworth a implementar. O código utilizado para o efeito foi:

```
[n_butter,Wn_butter] = buttord(0.8e6/(fa/2),1.4e6/(fa/2),3,50);
[B_butter,A_butter] = butter(n_butter, Wn_butter);
```

Em que a função *buttord* calcula a ordem do filtro necessária para atingir a atenuação especificada (50 dB) e a função *butter* forma os coeficientes do filtro Butterworth a utilizar.

Ou seja, os parâmetros do filtro são:

- Banda de Passagem: 0 até 0,8 MHz;
- Banda de corte: apartir dos 1,4 MHz;
- Atenuação: 50 dB;
- Ordem do filtro: 11;

De salientar que é necessário manter uma relação de compromisso entre a largura de banda de transição e atenuação do filtro. Ou seja, diminuindo uma terá de se diminuir também a outra. Nota-se também que as ordens dos filtros envolvidos são muito inferiores às consideradas para os filtros FIR.

3. Filtros elípticos

Tal como no caso dos filtros Butterworth, também aqui calculou-se a ordem com base nas bandas (passagem e corte) e atenuação pretendidas. E mais uma vez fazendo uso das funções disponibilizadas pela ferramenta *Matlab* calculou-se a ordem e os parâmetros e posteriormente criou-se o filtro.

O código utilizado foi:

```
[n_ellip, Wp_ellip] = ellipord(0.9e6/(fa/2),1.5e6/(fa/2),3,80);
[B_ellip,A_ellip] = ellip(n_ellip, 3, 80, Wp_ellip);
```

A função *ellipord* calcula a ordem do filtro necessária para atingir a atenuação especificada (80 dB) e a função *ellip* forma os coeficientes do filtro elíptico a utilizar.

As especificações do filtro utilizado são então:

- Banda de Passagem: 0 até 0,9 MHz;
- Banda de corte: apartir dos 1,5 MHz;
- Atenuação: 80 dB;

- Ordem do filtro: 6;

Também aqui é necessário manter uma relação de compromisso entre a largura de banda de transição e a atenuação do filtro. As especificações acima foram, de entre as estudadas, as que conduziram a melhores resultados.

3.3.3 Parâmetros de entrada da simulação

Para efeitos de simulação foram considerados os seguintes pressupostos abaixo enunciados.

- Taxa de transmissão do sinal = 500 kHz: Este valor é referido na norma EN 12253 como o ritmo de transmissão que deverá ser aplicado ao sinal;
- Taxa de transmissão do sinal codificado = 1 MHz: Depois de codificado o sinal passa a ter uma largura de banda que será o dobro da inicial (500 kHz), devido a cada bit ser codificado em dois (FM0);
- Frequência de amostragem = 20 MHz , 10 MHz e 5 MHz;
- $M \approx 0.8 / V_{\max} = 1V$ e $V_{\min} = 0,1V$;
- Número de bits de informação = 1000;
- Número de bits do preâmbulo = 16.

3.3.4 Cumprimento das máscaras espectrais sem DAC

Nesta simulação faz-se a comparação entre uma implementação utilizando filtros IIR (Butterworth e elíptico) e utilizando filtros FIR. O objetivo é concluir quais os filtros que oferecem uma melhor atenuação nas zonas do espectro referidas na tabela 3.1.

As tabelas abaixo apresentadas contêm os valores normalizados à potência máxima do sinal para três casos distintos. No primeiro o sinal não foi sujeito a nenhuma filtragem (banda base), no segundo considera-se o sinal depois de passar por um filtro FIR, Butterworth ou elíptico, enquanto que no terceiro considera-se o caso em que o sinal passou por uma cascata de filtros FIR, Butterworth ou elípticos.

Análise de filtros IIR

Começa-se por analisar qual o comportamento do espectro do sinal quando filtrado por um filtro IIR. Os filtros utilizados são os já explicados e detalhados anteriormente no trabalho.

Frequências	Largura de banda	Máscaras espectrais	Banda Base	Filtro elíptico		Filtro Butterworth	
				1 filtro	2 filtros	1 filtro	4 filtros
MHz	kHz	dB	dB	dB	dB	dB	dB
0	1000	0	0	0	0	0	0
1	62,5	-60	-68,6	-58,6	-64,3	-64,9	-69
1,5	500	-60	-18,9	-55,2	-63,1	-57,2	-69
2	500	-60	-31,8	-58,3	-68,3	-62,7	-74,4
3	500	-80	-35,2	-62,1	-74,2	-66,3	-80
3,5	500	-80	-26	-63,4	-76,1	-67,6	-81,7
4	62,5	-80	-80,1	-73,5	-86,6	-77,6	-92,1
6	62,5	-80	-82,9	-76,3	-90,3	-80,4	-95,4
6,5	500	-80	-30,3	-67,8	-81,9	-71,9	-86,9
7	500	-80	-41,1	-68,2	-82,4	-72,3	-87,3

Tabela 3.2: Comparação entre filtros IIR (elíptico e Butterworth)

Com a aplicação de uma filtragem recorrendo a filtros IIR em cascata é possível atingir-se os níveis de atenuação exigidos, como se pode ver pela tabela 3.2, para o caso de um filtro Butterworth. No caso do filtro elíptico os resultados estão muito próximos de atingir os objetivos, porém não se conseguiu um bom compromisso entre a largura da banda de passagem e a atenuação imposta pelo filtro por forma a cumprir as máscaras espectrais exigidas.

Análise filtro FIR com variação da frequência de amostragem

Nesta experiência usou-se um filtro FIR de ordem 100 e considerou-se um sinal amostrado a 20 MHz, 10 MHz ou 5 MHz. Fez-se uso do filtro FIR explicado anteriormente. O propósito desta simulação será estudar qual o comportamento do espectro considerando diferentes frequências de amostragem.

Frequências	Largura de banda	Máscaras espectrais	Banda base	Filtro FIR 100 coeficientes			
				20 MHz	10 MHz	5 MHz	4 filtros 20 MHz
MHz	kHz	dB	dB	dB	dB	dB	dB
0	1000	0	0	0	0	0	0
1	62,5	-60	-68,6	-71,3	-62,9	-61,8	-60,5
1,5	500	-60	-18,9	-52,5	-56,5	-55,5	-55,9
2	500	-60	-31,8	-63,8	-59,2	-57,3	-58,6
3	500	-80	-35,2	-66,9	-62,2	-57,3	-62,2
3,5	500	-80	-26	-59,6	-63	-55,5	-63,4
4	62,5	-80	-80,1	-95,6	-72,7	-61,8	-73,5
6	62,5	-80	-82,9	-97,7	-72,7	-61,8	-76,3
6,5	500	-80	-30,3	-64	-63	-55,5	-67,7
7	500	-80	-41,1	-72,3	-62,2	-57,3	-68,1

Tabela 3.3: Variação da frequência de amostragem num filtro FIR

Como é visível da tabela 3.3 não há grandes diferenças entre os resultados obtidos aplicando diferentes frequências de amostragem.

Deve-se salientar a presença de *aliasing* para as frequências de amostragem de 5 MHz e 10 MHz. Tal deve-se a se estar a considerar uma banda de frequências de interesse de 7 MHz. Logo de forma a eliminar a presença de *aliasing* seria necessário aplicar uma frequência de amostragem igual ou superior a 14 MHz, o dobro da frequência máxima das bandas em estudo (Teorema de *Nyquist*). Contudo tal não é prejudicial ao trabalho que se está a desenvolver, pois a zona correspondente à informação válida do sinal que se está a enviar encontra-se na zona dos 0 até 1 MHz, em banda base. A frequência de amostragem aplicada (10 MHz e 5 MHz) é por isso suficiente (10 e 5 vezes maior que a frequência do sinal, respetivamente).

Todavia a presença de *aliasing* ajuda a explicar os piores resultados obtidos para uma frequência de amostragem de 5 MHz nas bandas de frequências mais elevadas, pois irá existir sobreposição do espectro. Ou seja, o espectro irá repetir-se de 2,5 MHz em 2,5 MHz, logo as frequências menos atenuadas vão-se sobrepor às mais atenuadas, aumentando a potência.

Também no caso do filtro FIR simulou-se uma cascata de quatro filtros. Na tabela 3.3 está o resultado relativo à frequência de amostragem de 20 MHz. Não é relevante apresentar para todas as frequências pois os resultados são semelhantes. Verifica-se, ao contrário do sucedido com os filtros IIR que não há uma evolução na atenuação do sinal com a aplicação de vários filtros em cascata, apesar da resposta em frequência melhorar como é visível na figura 3.5 (a resposta que apresenta uma maior atenuação na banda de corte corresponde à cascata de 4 filtros). A razão para tais resultados estará possivelmente relacionada com os arredondamentos internos que são feitos. Os coeficientes calculados para os filtros têm valores que chegam a apresentar uma variação de três ordens de grandeza entre eles. Ao se utilizar uma representação em vírgula flutuante (*bits significativos * base^{expoente}*) poderão surgir erros de arredondamento. Erros esses devidos aos bits significativos considerados na representação poderem nuns casos apresentar grande parte da informação acerca do valor do coeficiente e noutros cortarem parte dessa informação. Outra explicação poderá estar relacionada com a maneira como é efetuada a *fft* por parte do *Matlab*.

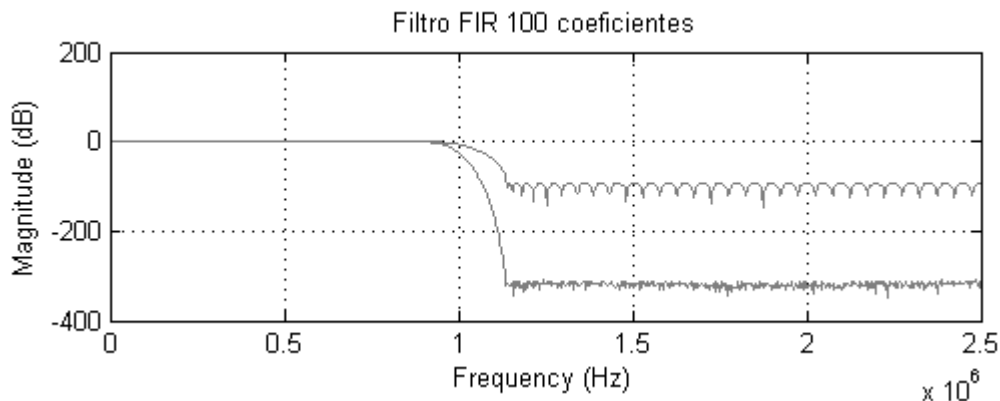


Figura 3.5: Resposta em frequência de um filtro FIR e uma cascata de 4 filtros FIR

Comparação dos gráficos temporais do sinal antes e depois de filtrado

Nesta secção ilustra-se a representação temporal do sinal antes e depois de tratado por um filtro FIR, Butterworth, ou elíptico. Consideram-se os casos que apresentaram melhores

resultados das tabelas 3.2 e 3.3.

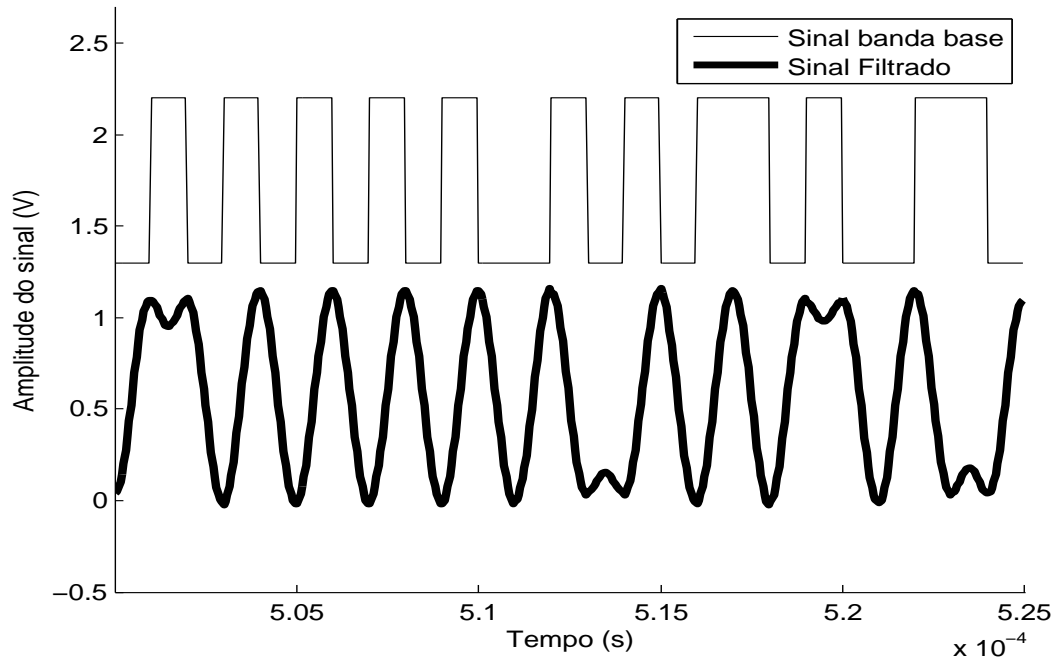


Figura 3.6: Diagramas temporais relativos ao filtro FIR

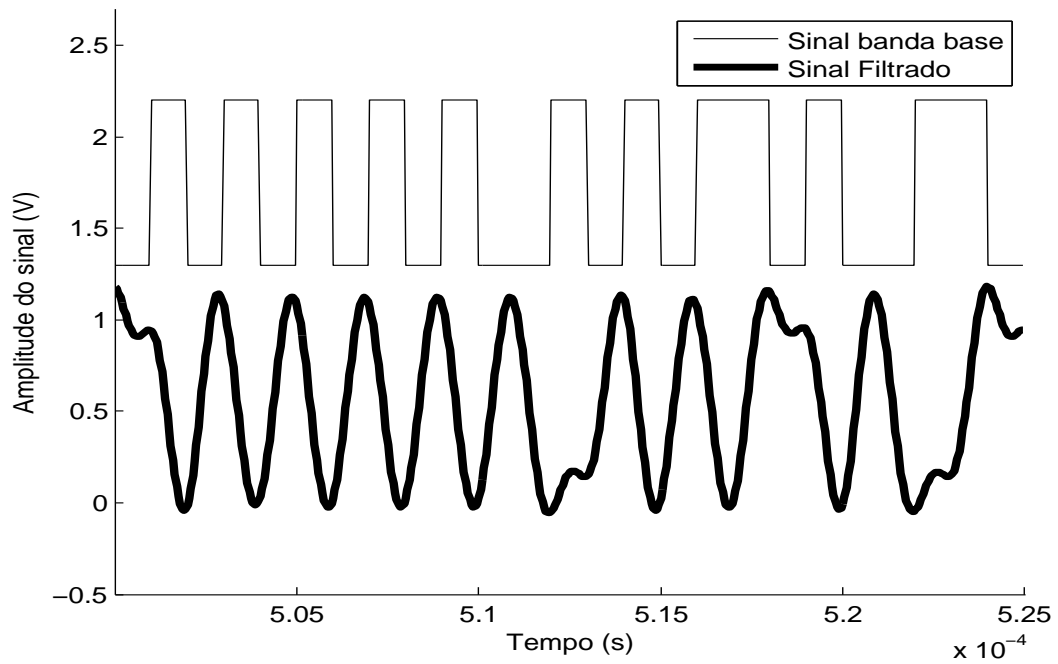


Figura 3.7: Diagramas temporais relativos ao filtro Butterworth

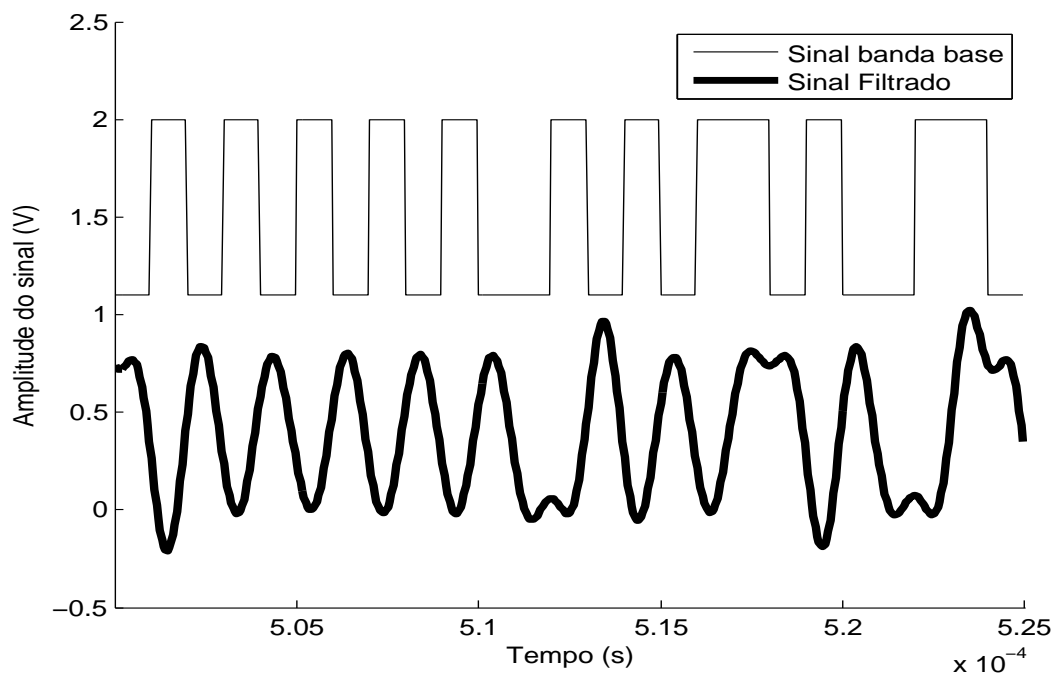


Figura 3.8: Diagramas temporais relativos ao filtro elíptico

Para os três diferentes casos o sinal há saída dos filtros apresenta uma forma idêntica à de entrada, ou seja, não há uma distorção significativa, apenas o normal “arredondar” do sinal devido à filtragem. Das figuras 3.6, 3.7 e 3.8 retira-se que o sinal à saída do filtro FIR apresenta uma melhor forma, quando comparado com os sinais obtidos à saída dos filtros IIR. Todavia o filtro FIR introduz um maior atraso ao sinal.

3.3.5 Cumprimento das máscaras espectrais com DAC

A passagem de digital para analógico é um dos aspetos do trabalho a que se deve dar especial atenção. A adição de uma DAC ao sistema implica uma maior quantificação do sinal, logo introdução de ruído, isto porque o sinal antes de ser apresentado à entrada da DAC encontra-se quantificado num maior número de bits. Há por isso necessidade de estudar se esse ruído irá prejudicar o trabalho já feito pelo filtro digital posto a montante. Além disso, o facto de se tratar de um módulo analógico impõe um melhor estudo antes da implementação, pois a alteração de um qualquer aspeto envolve a construção de um novo módulo. As simulações efetuadas terão como objetivo estudar o impacto que a quantificação terá no sinal, considerando DACs de 8, 10 e 12 bits e uma frequência de amostragem de 20 MHz.

Optou-se agora pela utilização de gráficos para apresentar os resultados das simulações, pois, nesta fase, dá-se mais ênfase à diferença entre os sinais após a sua quantificação pelas várias DAC. Pretende-se assim saber se a degradação introduzida é significativa e se existe alguma evolução com a utilização de um número maior de bits.

A simulação consiste na quantificação do sinal à saída do filtro em 256, 1024 ou 4096 níveis diferentes de forma a simular a quantificação efetuada por uma DAC. São apresentados para cada um dos filtros cinco casos, ou seja, comparou-se o sinal sem nenhuma filtragem, com o

signal depois de passar num filtro e com este depois de passar pelo conversor digital-analógico de '8', '10' ou '12' bits.

Com base na tabela 3.3 escolheu-se, para quantificação, o sinal tratado por apenas um filtro FIR. Tomou-se esta opção pois quanto maior a atenuação do sinal à saída do filtro mais se notará o efeito do ruído de quantificação introduzido pela DAC.

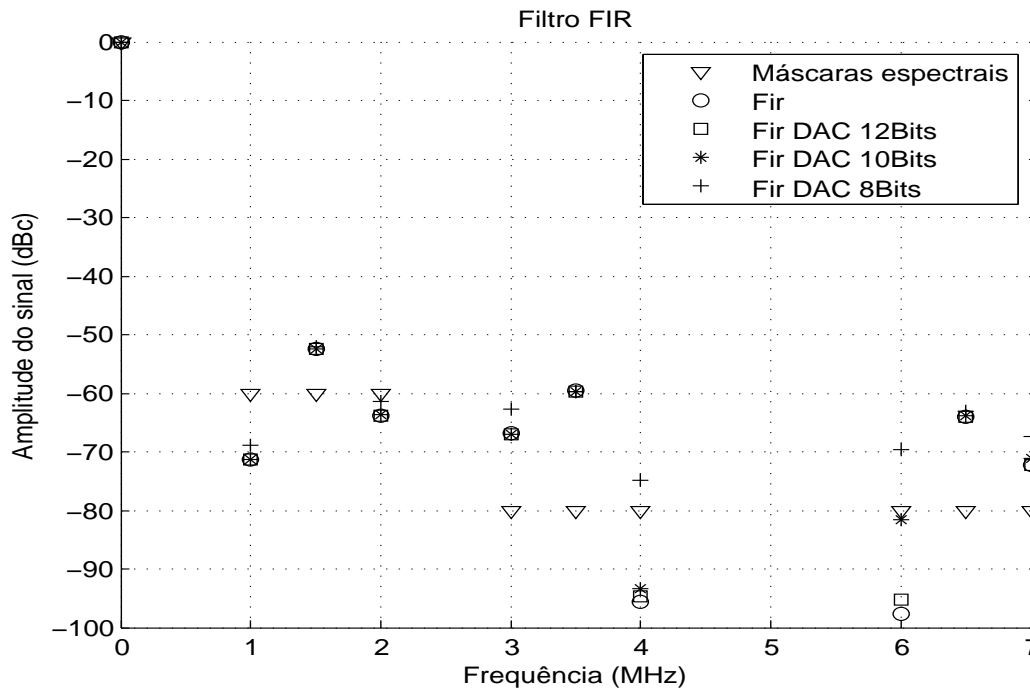


Figura 3.9: Ruído de quantificação (FIR)

Do estudo da figura acima infere-se que a aplicação de uma DAC de 8 bits irá introduzir uma quantidade de ruído de quantificação significativo, prejudicando o trabalho efetuado pelo filtro para se cumprir as máscaras espectrais. Já a aplicação de uma DAC de 10 ou 12 bits não terá um impacto muito negativo sobre o sinal, contudo a quantificação de 12 bits é a que apresenta melhores resultados, sendo por isso preferível.

Também para os dois gráficos ilustrados nas figuras 3.10 e 3.11 seguiu-se a lógica adotada no caso anterior. Ou seja, nestes dois casos considerou-se, com base na tabela 3.2, a cascata de quatro filtros (Butterworth) e dois filtros (elíptico).

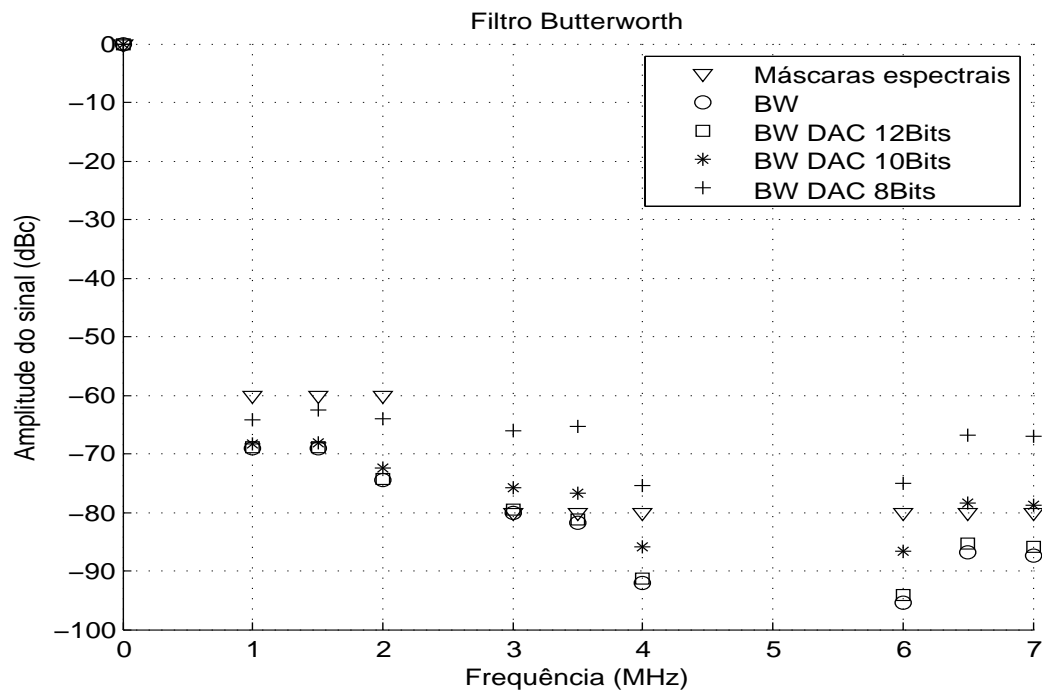


Figura 3.10: Ruído de quantificação (Butterworth)

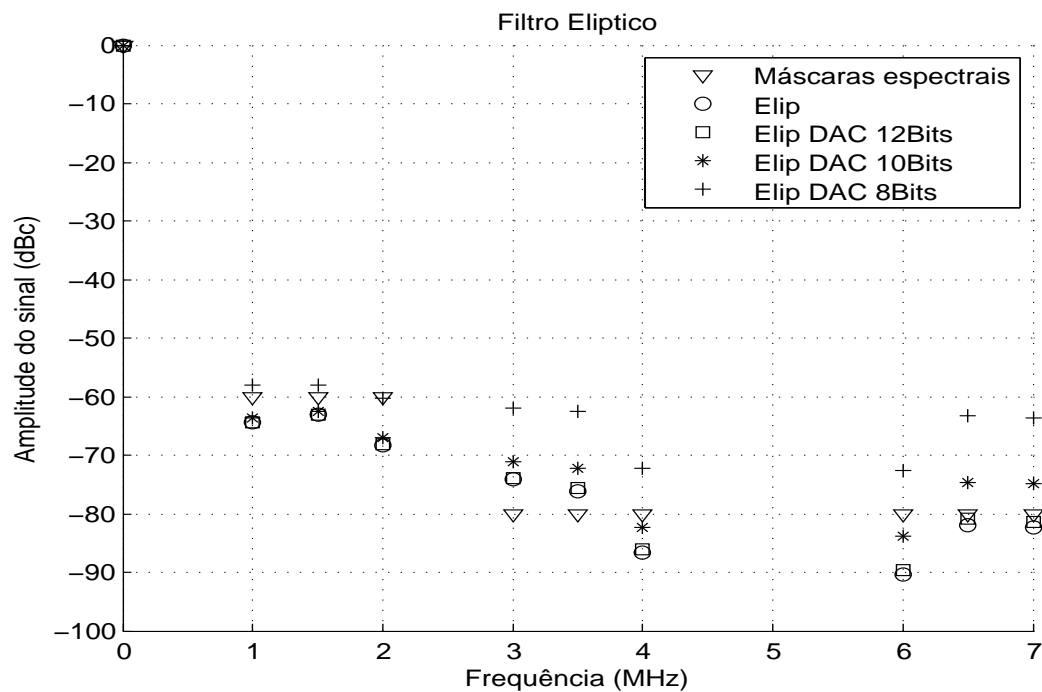


Figura 3.11: Ruído de quantificação (Elíptico)

No caso de uma filtragem recorrendo a filtros IIR, ao contrário do caso de um filtro

FIR, tanto a quantificação de 8 bits como a de 10 bits introduzem ruído de quantificação significativo. Nestes casos será necessário uma quantificação de 12 bits ou superior.

Conclui-se que a quantificação parece ser mais prejudicial no caso dos filtros IIR em comparação com os filtros FIR.

Ruído de quantificação esperado

Além das simulações já efetuadas para verificar qual o impacto da quantificação no desempenho do sistema, pretende-se agora quantificar qual a SQNR que se tem à saída de cada um dos conversores digital-analógico considerados (8, 10 e 12 bits).

A SQNR foi calculada recorrendo à expressão 2.4 já explicada no capítulo 2. Os resultados obtidos encontram-se na tabela 3.4.

	8 bits	10 bits	12 bits	24 bits
SQNR (dB)	48,16	60,21	72,23	144,49

Tabela 3.4: SQNR

A tabela 3.4 mostra a SQNR esperada consoante o ruído de quantificação introduzido. Lembrando os valores que obtivemos nas tabelas 3.2 e 3.3, e comparando-os com os valores agora calculados, chega-se à conclusão que a margem que é garantida pelos conversores de 8, 10 e 12 bits não é elevada o suficiente por forma a não prejudicar os níveis de atenuação máximos obtidos pela aplicação dos filtros. Todavia os gráficos 3.9, 3.10 e 3.11 mostram que a quantificação recorrendo a 12 bits não introduz ruído significativo ao sinal para nenhum dos casos estudados.

A SQNR calculada para 24 bits foi feita pois verificou-se que na implementação de um filtro FIR em FPGA, a sua saída era representada com 24 bits. Pretendeu-se por isso verificar o quanto é prejudicada a SQNR com a quantificação feita pela DAC, em relação ao sinal que se teria se não houvesse a quantificação imposta pela DAC.

3.3.6 Conclusões

As conclusões gerais a tirar da bateria de simulações efetuadas serão:

- Filtro digital:

Apesar de se conseguir atingir o cumprimento das máscaras espectrais com utilização de um filtro digital do tipo IIR, a opção de qual o filtro digital a implementar recairá sobre um filtro do tipo FIR. Optar-se-á por esta implementação devido a dois aspetos. O primeiro está relacionado com as vantagens, já discutidas, oferecidas pelos filtros FIR em comparação com os IIR (exemplo: facilidade de implementação devido à disponibilização de *cores*). O segundo advém da necessidade de se colocar um filtro analógico à saída da DAC para reconstrução do sinal, que acabará por introduzir uma certa atenuação no sinal contribuindo para o cumprimento das máscaras espectrais. A estratégia para o cumprimento das máscaras espectrais passará então pela utilização de uma primeira filtragem a nível digital seguida, posteriormente, de uma filtragem ao nível analógico (daí a maior importância do filtro analógico!).

- DAC:

O estudo levado a cabo nestas simulações revelou que se deverá optar pela escolha de uma DAC de 12 bits, por forma a diminuir o ruído de quantificação introduzido no sinal. Esta escolha seria sempre a mais aconselhável pois sendo este um projeto de teste é importante manter um certo grau de liberdade de maneira a permitir um melhor estudo do comportamento global do sistema. Em relação à frequência de funcionamento escolher-se-á 20 MHz. Isto porque é uma frequência suficiente grande de modo a providenciar uma certa liberdade e evitar a ocorrência de *aliasing*, mas não tão grande ao ponto de se estar a desperdiçar desempenho.

- NOTA: A incorência que se verificou nas simulações de um filtro FIR simples e uma cascata de filtros FIR não representa problemas de maior para as conclusões que se pretenderam retirar das simulações efetuadas. Ou seja, os resultados para um filtro FIR simples são fidedignos e embora os da cascata de filtros não o serem, a opção mais viável do ponto de vista da implementação na FPGA será a utilização de apenas um filtro. Isto porque haverá a utilização de um filtro analógico e assim também se necessitará de menos recursos, no que toca à FPGA. O objetivo do teste da cascata de filtros FIR seria observar se haveria melhoria na atenuação, infelizmente não foi possível tirar conclusão.

Capítulo 4

Implementação do emissor

4.1 Introdução

Discute-se agora a implementação do emissor que irá ser criado. Faz-se referência ao projeto modelado na FPGA, para efetuar o tratamento em banda base do sinal a enviar. E aborda-se também o *front-end* desenvolvido, mais concretamente o conversor digital - analógico, fazendo uma leve referência ao filtro passa-baixo analógico já existente que se irá utilizar.

4.2 Arquitetura do emissor

A figura 4.1 ilustra o diagrama de blocos do sistema a ser modelado na FPGA. Optou-se por utilizar vários blocos pouco complexos de modo a tornar a implementação mais simples, intuitiva e facilitar a reconfiguração do sistema, se tal for necessário. Ainda em relação à figura 4.1, há a salientar que nas linhas de interligação entre os diferentes blocos encontra-se indicado o número de bits que se utiliza para representar um símbolo. Ou seja, é enviado um símbolo de cada vez, mas o número de bits necessários para identificar o símbolo vai variando.

De seguida irá proceder-se a uma explicação de cada um dos blocos.

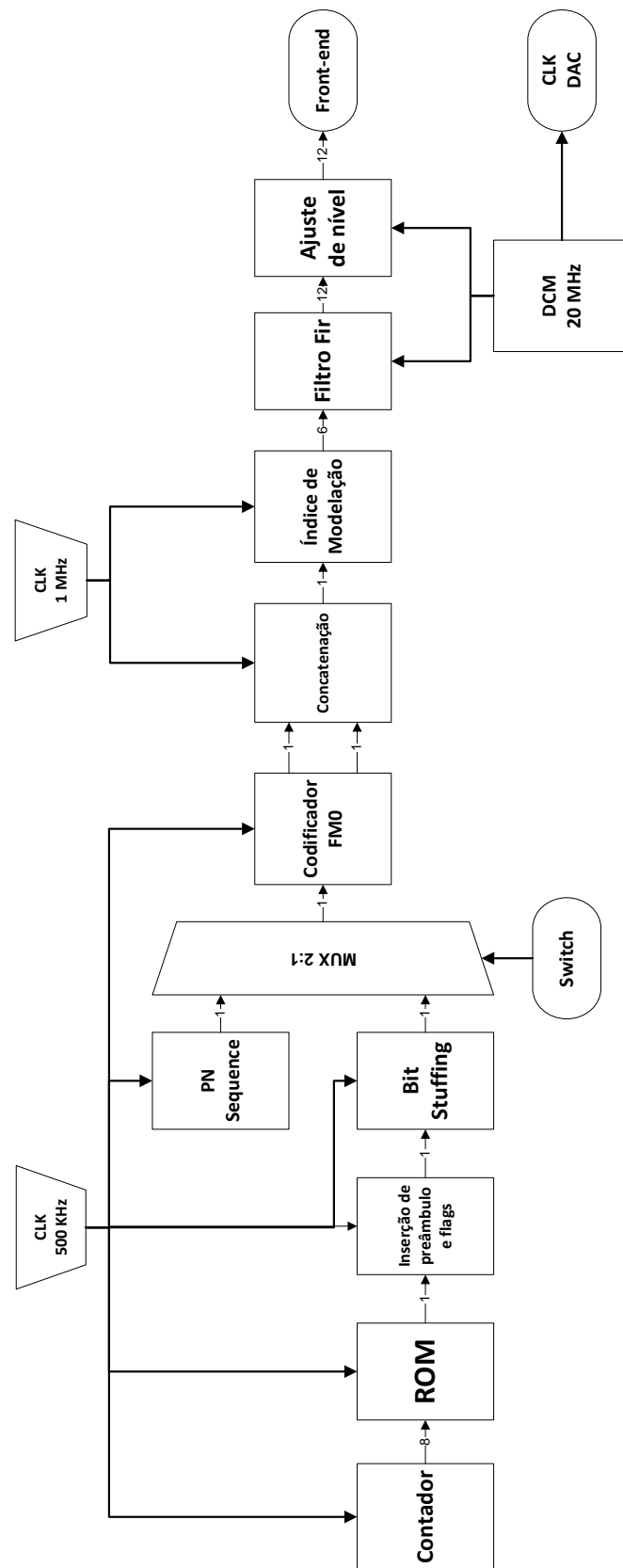


Figura 4.1: Diagrama de blocos do emissor referente ao interior da FPGA

1. *Clocks* 500 kHz, 1 MHz e 20 MHz;

São utilizados três *clocks* diferentes. O de 500 kHz corresponde ao ritmo de transmissão do sinal indicado na norma EN12253. Já o de 1 MHz é utilizado para o sinal codificado, pois neste caso cada bit é codificado em 2 sendo assim necessário duplicar a frequência do sinal. O *clock* de 20 MHz diz respeito à frequência de amostragem que irá ser utilizada pelo filtro e também na DAC. Para a geração desta frequência recorreu-se a um DCM fazendo uso de um *IP-Core* já disponibilizado pelo fabricante.

Os DCMs são utilizados para manipular os sinais de *clock* e permitem:

- Multiplicar e dividir um sinal de entrada de *clock*;
- Recondicionar um sinal de *clock* de modo a apresentar, por exemplo, um *duty cycle* de 50%;
- Controlar a fase do clock;
- Diminuir o *clock skew*.

Os DCMs fazem uso de linhas de controlo existentes na FPGA para o encaminhamento dos sinais de *clock*.

No caso de um *clock* de 20 MHz é aconselhável fazer uso destes blocos.

2. Contador e ROM;

Na abordagem adotada para este sistema pressupõe-se que a informação a ser transmitida tal como o código CRC estão já criados e armazenados no interior de uma ROM.

O contador destina-se a indexar as posições de memória de forma a que a informação seja apresentada à saída corretamente.

Estes dois blocos são controlados por o relógio de 500 kHz e a saída da memória ROM é em série com apenas 1 bit.

3. Inserção do preâmbulo e *flags*;

Neste bloco faz-se a introdução do preâmbulo referido na norma EN 12253 e também se introduzem a *flags* que sinalizam o início e fim de uma trama e que são mencionadas na norma EN 12795.

4. BitStuffing;

O *bit stuffing* referido na norma EN 12795 é efetuado nesta unidade. Um aspeto importante deste bloco é que como todo o sistema funciona em tempo real, quando há necessidade de se introduzir um bit de *stuff*, existe a necessidade de parar toda a cadeia a montante de maneira a não haver perdas de informação. Esse problema é resolvido recorrendo a sinais de controlo e também a blocos de atraso pois cada bloco a montante introduz um atraso de um ciclo de relógio que terá de ser compensado aquando da paragem do sistema.

5. PN Sequence:

Este bloco destina-se à criação de uma sequência de bits aleatória, visando testar o funcionamento do emissor. Esta sequência respeita o protocolo dPMR, protocolo esse

criado pelo ETSI e que tem o objetivo de garantir a interoperabilidade entre os diversos aparelhos usados em comunicações radio móveis.

A informação transmitida tem então o aspeto ilustrado na figura 4.2.

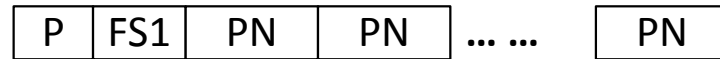


Figura 4.2: Exemplo de uma sequência pseudo-aleatória

Onde:

- P : Corresponde ao preâmbulo que deverá ter um mínimo de 72 bits. Em hexadecimal terá a forma: $5F5F5F5F5F5F5F5F5F$.
- FS1 : Diz respeito à sequência de sincronização que terá um comprimento de 48 bits e terá a forma, em hexadecimal: $57FF5F75D577$.
- PN : É a sequência aleatória de 511 bits. A criação desta sequência será efetuada recorrendo a um *shift-register* idêntico ao mostrado na figura 4.3.

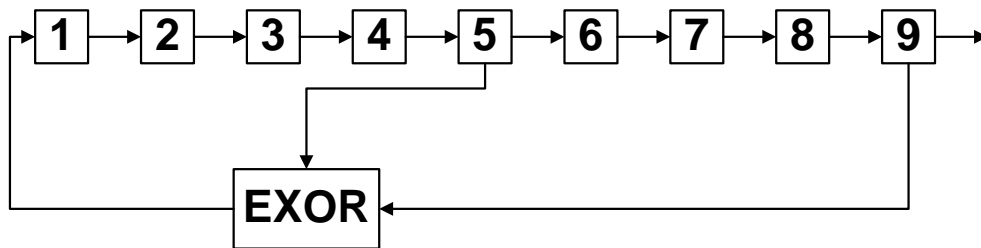


Figura 4.3: Shift-register implementado para criar a sequência pseudo-aleatória

6. Codificação FM0;

Como o próprio nome indica aqui é feita a codificação do sinal. Esta codificação segue as regras já enunciadas do FM0. De referir que se optou por fazer uma saída em paralelo ao invés de em série, por forma a manter o bloco o mais simples possível, seguindo a ideologia em cima discutida.

7. Multiplexer e Concatenação;

O multiplexer permite efetuar a escolha de qual a mensagem que se quer enviar. Poderá optar-se pela mensagem de informação real, contida na ROM, ou então pela mensagem pseudo-aleatória. Essa escolha é efetuada em tempo real recorrendo apenas à comutação de um *switch* embutido na placa de desenvolvimento, não sendo por isso necessário interromper o funcionamento do sistema.

O bloco de concatenação destina-se apenas a agrupar a saída do codificador, de forma a que se passe a ter uma saída em série.

8. Índice de Modulação

O sinal a ser enviado deverá apresentar um índice de modulação entre 50% e 90% dado pela equação 2.1. O pior caso, isto é, em que há maior energia do sinal espalhada pelo espectro, ocorre para um índice de modelação de 90%. Será aproximadamente este o índice que se tentará impor.

Este bloco o que fará então será a associação entre o 0 e o nível baixo e o 1 e o nível alto, em que os níveis alto e baixo têm os valores de 1 e aproximadamente 0.1, respetivamente, de maneira a ter-se um índice de modulação pretendido. A representação de cada bit de informação será agora feita através de 6 bits em vez de apenas 1 bit. Dos 6 bits considerados 1 representa a parte inteira ao passo que os outros 5 dizem respeito à parte decimal. Ou seja:

- Representação 1 volt: $1,00000_2 = 1$;
- Representação 0,1 volt: $0,00011_2 = 0,0938$;
- Índice de modulação: $m \approx 0,82$, recorrendo à equação 2.1;

9. Filtro Fir;

Aproveitando o *core* disponibilizado pelo fabricante no *software* de apoio à configuração da FPGA, criou-se um filtro FIR. O *core* que possibilita a construção do tal filtro denomina-se por *FIR Compiler* e disponibiliza vários tipos de configurações [6]. No caso deste emissor implementou-se um filtro FIR simples, sendo que as suas principais características são:

- Entradas *unsigned*;
- Parte inteira (1 bit) e parte decimal (5 bits);
- Saída *signed*, com o mesmo número de bits que a DAC utilizada (12, 10 ou 8 bits);
- Arredondamento para o número par mais próximo.

O vetor dos coeficientes foi criado utilizando a ferramenta *FDA tool* disponível no *MatLab*. A figura 4.4 corresponde à janela de configuração do filtro, disponibilizada pela *FDA tool*.

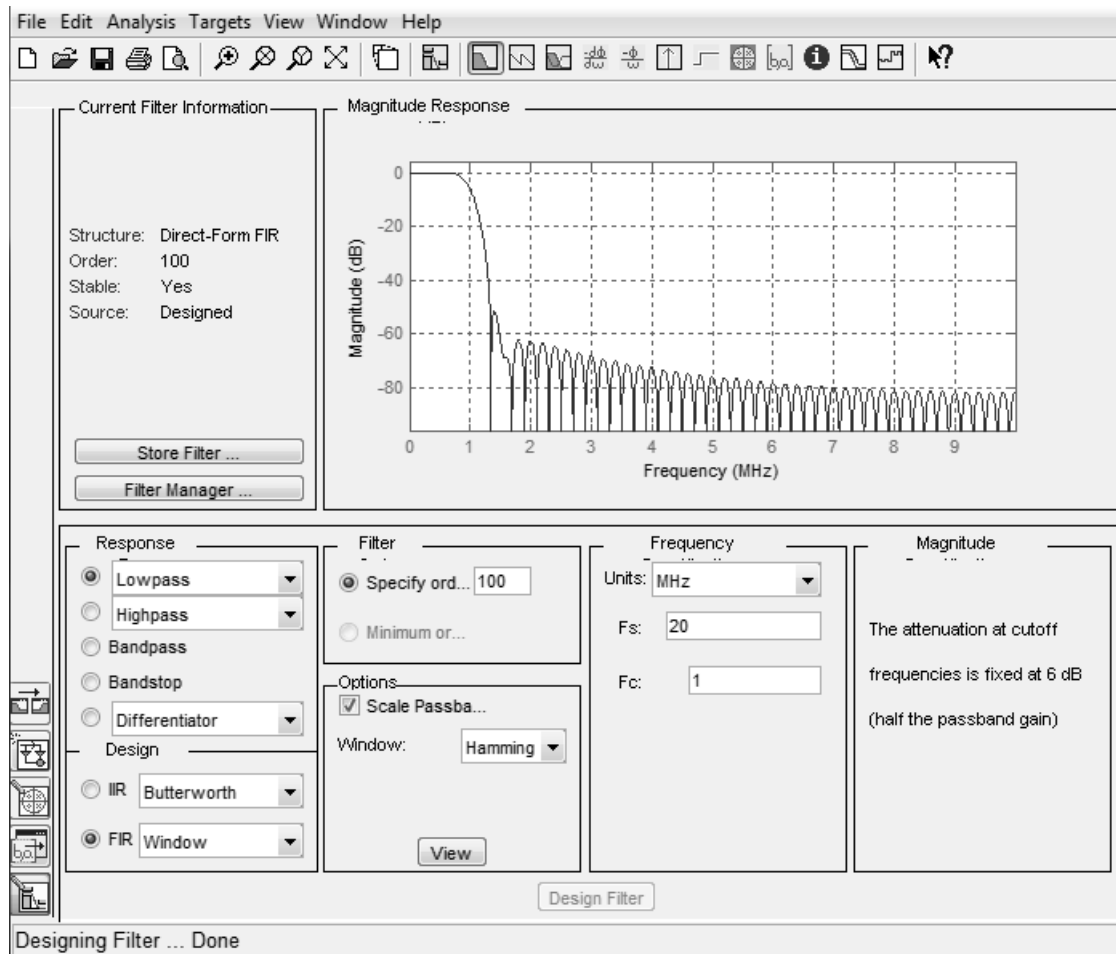


Figura 4.4: Janela de configuração (*FDA tool*)

Como se pode constatar pela figura criou-se um filtro com as seguintes especificações:

- Filtro FIR passa-baixo;
- Frequência de corte: 1 MHz;
- Frequência de amostragem: 20 MHz;
- Ordem do filtro: 100;
- Largura da banda de passagem: Definida pela *FDA tool*, somente se determinou que deveria respeitar as características de uma janela de *Hamming*. O projeto recorrendo ao método de janelas de *Hamming* mostrou em comparação com outros ser um dos que melhores resultados apresentou, além de exigir a definição de menos parâmetros tornando o seu projeto mais simples.

10. Ajuste de nível;

A necessidade de utilização deste bloco advém do funcionamento muito característico do *front-end* que engloba o conversor digital-analógico. O *front-end* foi desenhado para

funcionar em modo bipolar, disponibilizando assim uma saída que poderá ser positiva ou negativa. Contudo a entrada da DAC não é feita em complemento para dois, sendo que quando o bit mais significativo é '0' a saída é negativa e quando é '1' a saída é positiva. Há assim necessidade de ajustar a saída do filtro digital para se ajustar ao funcionamento do *front-end*. A solução implementada foi somar metade da escala ao valor que se tem à saída do filtro, eliminando assim o bit de sinal (bit mais significativo).

11. *Front-end*

O *Front-end* utilizado irá variar consoante os testes efetuados. Utilizaram-se uma DAC e um filtro analógico, aos quais se faz uma referência mais pormenorizada à frente no trabalho. Além destes foi utilizado um *mixer*, um oscilador de 5,8 GHz e uma antena desenhada para comunicações DSRC.

4.3 Modelo digital

A placa de desenvolvimento que servirá de apoio ao projeto é uma *ML505* (figura 4.5). Estas placas englobam, além dos vários periféricos, uma FPGA *virtex-5* modelo *XC5VLX50T-1FFG1136* do fabricante *Xilinx's*. A configuração da FPGA é feita recorrendo ao *software* disponibilizado pela *Xilinx's* denominado *Xilinx ISE design suite 12.1* utilizando linguagem VHDL. A comunicação entre o computador e a placa é assegurada por uma ligação USB - JTAG, respetivamente. O JTAG permite, através do *Boundary scan testing*, testar se determinado conjunto de ICs estão a funcionar corretamente, possibilitando a configuração destes para a realização dos testes. Outras das vantagens da utilização do JTAG advém deste possibilitar a recolha de sinais do interior da placa para *debugging*. A aplicação usada no PC para fazer a comunicação foi o *Impact*, também disponibilizado pelo fabricante.

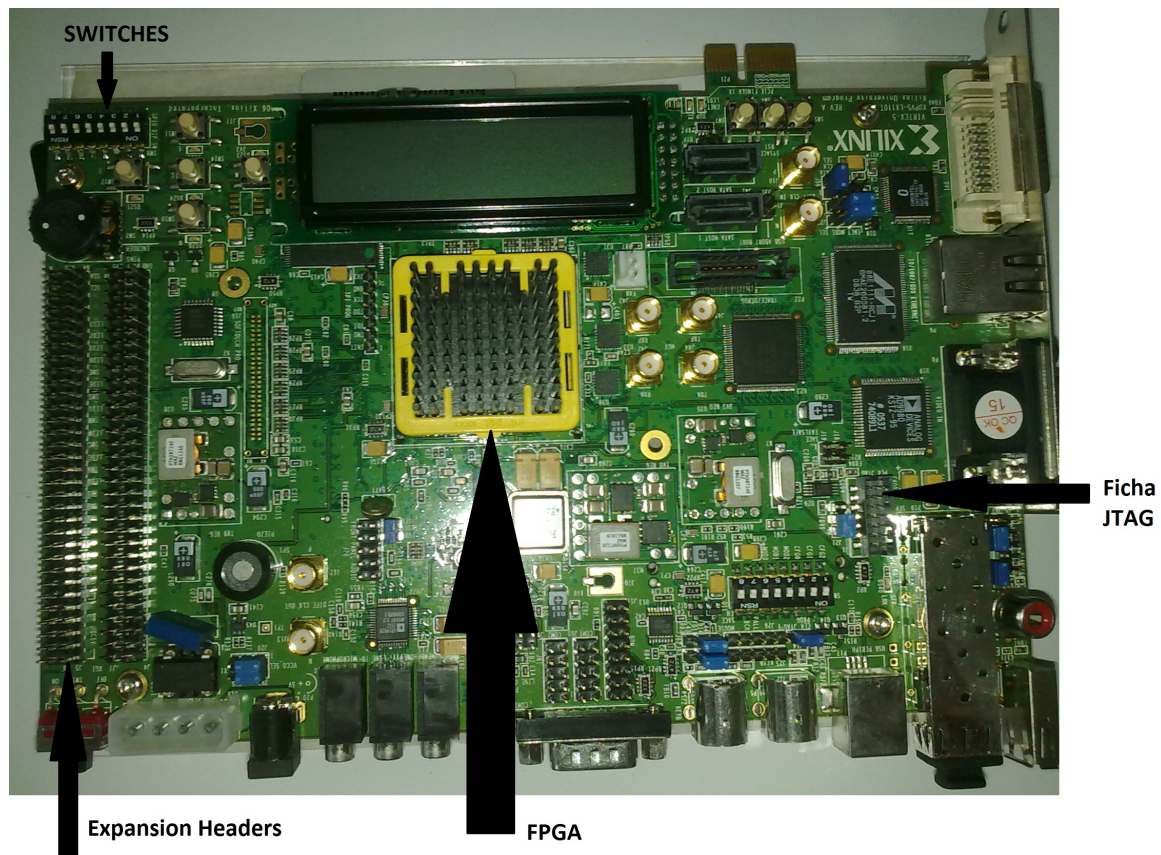


Figura 4.5: Placa de desenvolvimento

Recorrendo a estas ferramentas modelou-se a FPGA com a arquitetura acima descrita.

Na tabela 4.1 estão resumidos os recursos utilizados na implementação do projeto “emissor” considerando um filtro de 100 coeficientes, uma DAC de 12 bits e uma frequência de amostragem de 20 MHz, ou seja, o pior caso em termos de recursos utilizados.

A tabela apresentada é disponibilizada pelo *software* de desenvolvimento utilizado para modelar a FPGA. O *Xilinx ISE design suite 12.1*. A interpretação dos resultados mostrados na tabela fornecem informação acerca da ocupação total da FPGA. É uma preciosa ajuda para se escolher qual FPGA que pode ser escolhida para a implementação do projeto. Ou seja, no caso do presente projeto já se tem uma FPGA disponível, porém se não fosse esse o caso o *software* utilizado permite testar se dada implementação é possível de fazer em dada FPGA permitindo escolher qual a melhor FPGA a utilizar. Possibilita ao modelador escolher o melhor compromisso entre recursos utilizados e custos económicos.

Device Utilization Summary				
Slice Logic Utilization		Used	Available	Utilization
N ^o of Slice Registers		849	69120	1%
N ^o of Slice LUTs		1128	69120	1%
	N ^o used as logic	835	69120	1%
	N ^o used as memory	287	17920	1%
N ^o of occupied Slices		423	17280	2%
N ^o of LUT Flip-Flop pairs used		1155		
	N ^o with an unused Flip-Flop	306	1155	26%
	N ^o with an unused LUT	27	1155	2%
	N ^o of fully used LUT-FF Pairs	822	1155	71%
	N ^o of slice register sites lost to control set restrictions	100	69120	1%
N ^o of bounded IOBs		18	640	2%
	N ^o of LOCed IOBs	17	18	94%
N ^o of BlockRAM/FIFO		1	148	1%
	Total memory used (kB)	18	5328	1%
N ^o of BUFG/BUFGCTRLs		3	32	9%
N ^o of DCM_ADVs		1	12	8%
N ^o of DSP48Es		46	64	71%

Tabela 4.1: Recursos, da FPGA, utilizados

A resultados obtidos dão desde já conta da limitação existente quanto ao número de coeficientes e filtros que se poderá utilizar, pois tendo em atenção que as multiplicações são asseguradas pelos *DSP48Es* conclui-se que apenas pela utilização de um filtro FIR com 100 coeficientes ocupasse quase 3/4 dos *DSP48Es* existentes.

Em relação aos restantes recursos há uma boa margem de manobra, não derivando daí nenhum tipo de limitação à implementação do projeto. De notar que as percentagens relativas ao *N^o of LOCed IOBs* e ao *N^o of fully used LUT-FF Pairs* são relativas, isto é, não se considera o número total de blocos existentes no interior da FPGA. Nestes casos entrando com o número total de blocos existentes ter-se-ão as percentagens de 1.2% (*N^o of fully used LUT-FF Pairs*) e de 2% (*N^o of LOCed IOBs*), o que mostra que a capacidade da FPGA é suficiente.

4.4 Front-End Analógico

Tal como ilustrado na figura 3.1 terá que se acrescentar *hardware* extra à FPGA para se transmitir corretamente o sinal pretendido. Esse *hardware* extra trata-se de componentes analógicos e consiste no conversor digital-analógico, filtro passa-baixo analógico, um *mixer* e oscilador de 5,8 GHz, um amplificador, um filtro passa-banda e uma antena.

Faz-se agora referência a dois desses componentes: o módulo com o conversor digital-analógico e o filtro passa-baixo.

De notar que visto os testes de comunicação entre a RSU e OBU serem feitos em laboratório não há necessidade de amplificar e aplicar o filtro passa banda, pois a distância entre os equipamentos será muito curta e não existirão interferências externas significativas.

4.4.1 Conversor Digital-Analógico

No trabalho realizado apenas foi necessário a construção de raiz do módulo que contém o conversor digital-analógico.

A DAC foi escolhida a partir das simulações e de forma a oferecer um certo grau de liberdade ao sistema. Recaindo a escolha numa DAC com uma frequência de amostragem de aproximadamente 20 MHz e com 12 bits de quantificação.

A interligação entre o módulo analógico e a FPGA é feita pelos *XGI Expansion Headers* presentes na placa de desenvolvimento (figura 4.5). Estes *headers* permitem uma fácil e rápida conexão entre as duas unidades, além de disponibilizarem alimentações de 3,3 volt, 5 volt e *ground* necessárias ao módulo analógico. Também a necessidade de utilizar a mesma FPGA tanto para o emissor como para o recetor torna o número de *headers* existentes um fator importante na escolha deste meio de ligação.

A figura 4.6 apresenta o modelo simplificado do circuito construído para efetuar a conversão de dados de digital para analógico. Este circuito teve como base o circuito apresentado no datasheet da DAC [7].

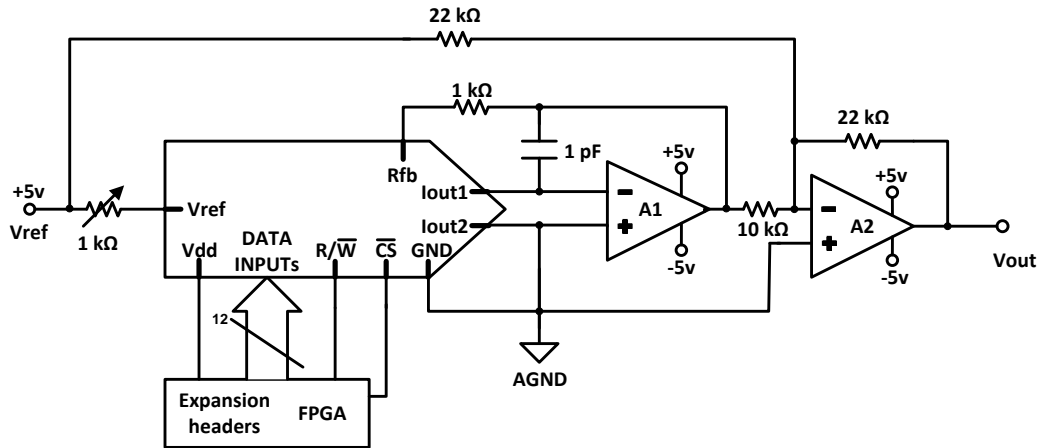


Figura 4.6: Modelo do conversor digital-analógico

Explicam-se agora algumas características do circuito acima ilustrado:

- A saída da DAC é feita em corrente. O amplificador 'A1' é por isso utilizado para fazer a conversão da saída de corrente para tensão.
- A saída final (V_{out}) é bipolar. Optou-se por essa implementação pois a saída do filtro digital assume por vezes valores negativos. A tensão de saída bipolar é conseguida através da

introdução do amplificador 'A2' e é dada pela equação:

$$V_{out} = V_{ref} * \frac{D}{2^{n-1}} - V_{ref} \quad (4.1)$$

Em que 'D' é o valor decimal presente à entrada da DAC e 'n' a sua resolução.

- Fonte de alimentação externa, de forma a obter-se alimentações simétricas necessárias para polarizar os amplificadores.
- Uso do potenciômetro para se poder ajustar o nível correspondente aos 0 volt, ou seja, fazer corresponder os 0 volt à saída ao valor de entrada da DAC 10000000000_2 .
- Níveis de tensão:
 - $V_{DD} = 3,3volt$, corresponde à tensão associada a um bit a 1 à entrada da DAC;
 - $V_{ref} = 5volt$. Tensão de referência da DAC.
- Entradas 'R/W' e 'CS'. A entrada 'R/W' indica se se pretende escrever ou ler da DAC. Como a escrita é ativa negativa e só se pretende escrever, o valor aí escrito será sempre um 0 lógico. A entrada 'CS' corresponde ao *clock* da DAC.

No anexo B são apresentados detalhes relativos aos componentes utilizados e ao PCB construído.

4.4.2 Filtro Analógico

Por existir um filtro analógico criado especificamente para comunicações DSRC, a construção deste não foi tida como objetivo do trabalho. Contudo convém fazer uma breve referência ao filtro utilizado.

O filtro analógico disponível é do tipo elíptico, passa-baixo e as suas características são:

- Oferece a melhor rejeição recorrendo à menor ordem;
- Apresenta *ripple* tanto na banda de passagem, como na banda de corte;
- Tem ordem par, 6. Ou seja, a resposta do filtro é sempre decrescente. Quanto maior a frequência maior a atenuação.

Levando em linha de conta que as máscaras espectrais são tanto mais rigorosas quanto mais afastada da portadora é a banda de interesse, um filtro de ordem par é o apropriado.

A figura 4.7 ilustra a resposta em frequência do filtro analógico utilizado.

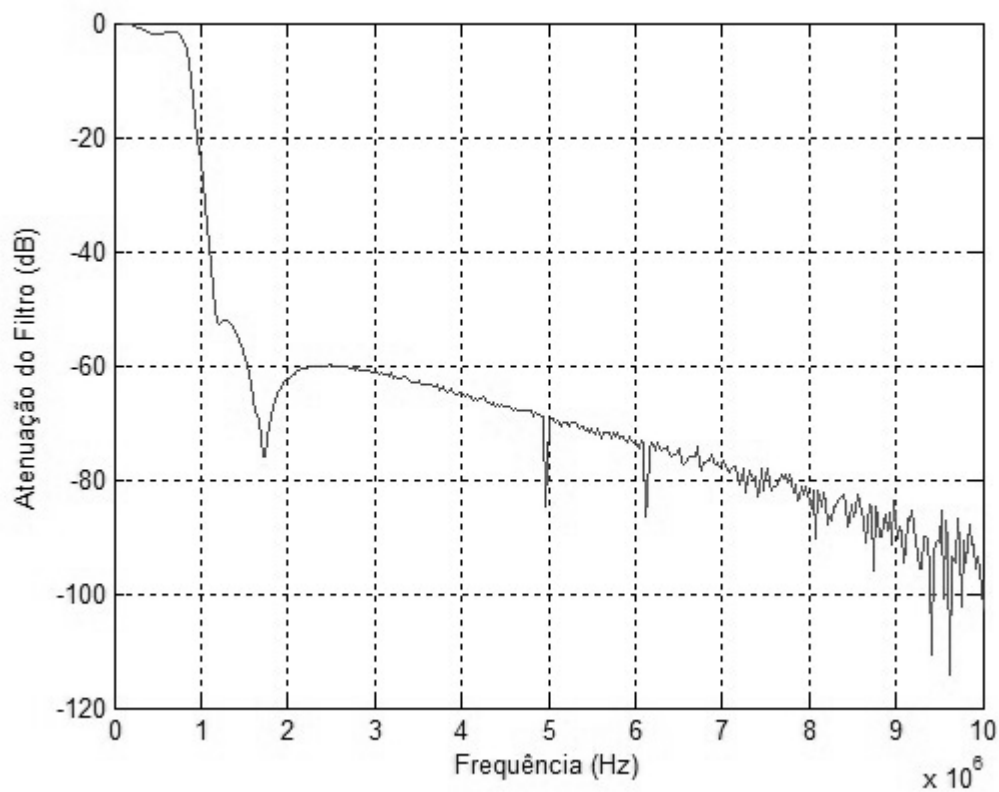


Figura 4.7: Espectro do filtro analógico

Como é facilmente observável pela figura, o filtro apresenta uma grande atenuação fora da banda de passagem, que aliada à atenuação já efetuada pelo filtro digital deverá ser suficiente para assegurar o cumprimento das máscaras espectrais tabeladas (Tabela 3.1). Ainda da figura 4.7 é possível observar que o filtro é bastante abrupto na zona de corte, apresentando já uma grande atenuação na frequência de corte de 1 MHz. É o preço a pagar por forma a se ter um filtro com uma grande atenuação, logo nas frequências mais baixas, e assim se conseguir cumprir os requisitos das máscaras espectrais. É preciso não esquecer que este filtro foi construído não entrando em linha de conta que se teria já um sinal filtrado a nível digital, como acontece neste trabalho. Será por isso de prever que as máscaras espectrais sejam cumpridas sem grandes dificuldades.

Capítulo 5

Validação experimental e resultados

5.1 Introdução

Procede-se agora à validação do emissor projetado.

Os testes experimentais efetuados recaem sobre três aspetos fundamentais:

1. Máscaras espectrais;
 - Verificar o desempenho do filtro FIR implementado;
 - Comprovar que pela aplicação do filtro analógico, a seguir ao conversor digital-analógico, se consegue efetivamente cumprir os limites de potência enunciados na norma relativa à camada física.
2. Diagramas de olho;
 - Recorrer-se-á aos diagramas de olho como forma de avaliar a qualidade do sinal que se tem à saída da DAC, antes e depois de passar por o filtro analógico.
3. Comunicação entre RSU e OBU.
 - Neste último teste é utilizado uma OBU comercial e o objetivo é que o emissor criado envie uma mensagem válida que seja detetada pelo equipamento comercial, desencadeando uma resposta do mesmo.

Pretende-se com estes testes comprovar que o emissor criado consegue enviar para um equipamento comercial, um sinal válido e reconhecível do ponto de vista do mesmo e das normas em vigor (camada física e parte da camada de ligação de dados).

5.2 Máscaras espectrais

O cumprimento das máscaras espectrais referido na norma da camada física do DSRC é um dos aspetos fundamentais do trabalho realizado. O objetivo desta secção será verificar se se consegue cumprir as referidas máscaras sem e/ou com o filtro analógico. Tal como verificar qual a diferença existente nas máscaras variando o número de bits de quantificação da DAC.

5.2.1 Testes e resultados

A figura 5.1 contém o material utilizado para a obtenção dos resultados.

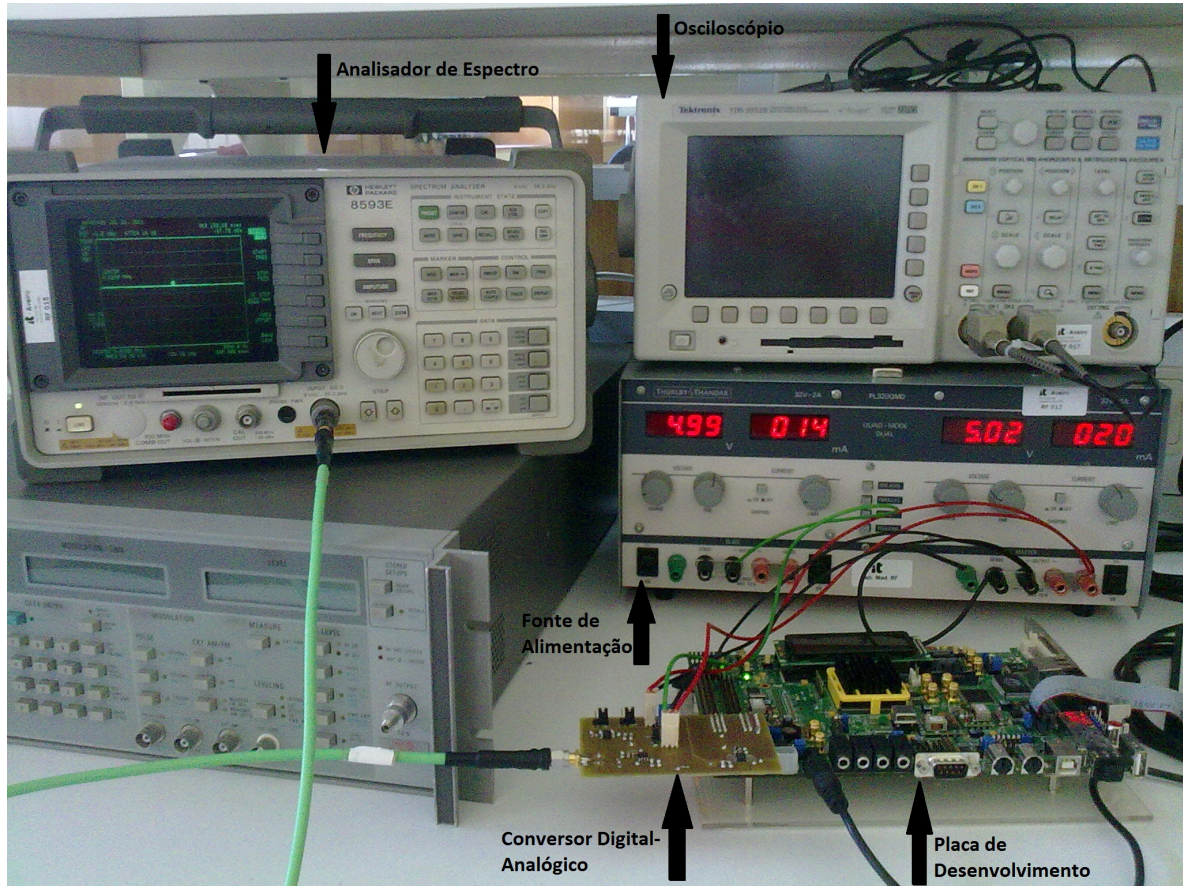


Figura 5.1: Equipamentos envolvidos na aquisição de resultados

Além dos dispositivos pertencentes ao emissor (FPGA e DAC) pode-se observar também os aparelhos eletrônicos que se utilizaram. São eles:

- Osciloscópio: permite confirmar o correto envio do sinal (forma, ruído, tempos de bit de informação).
- Analisador de espectro: utilizado para se calcular qual a potência presente em cada uma das bandas de interesse.
- Fonte de alimentação: necessária para providenciar a alimentação bipolar aos amplificadores responsáveis pela conversão da corrente de saída da DAC em tensão.

Na figura 5.2 estão representados mais pormenorizadamente os dispositivos que serão testados. Mais concretamente a FPGA, a DAC e o filtro analógico. A figura permite também observar como estes três componentes são interligados entre eles, ou seja, têm-se uma ligação por *expansion headers* entre a FPGA e a DAC e uma ligação por cabo coaxial entre a DAC e

o filtro analógico. Ligação essa que se repete para interligar o filtro analógico e o analisador de espectro.

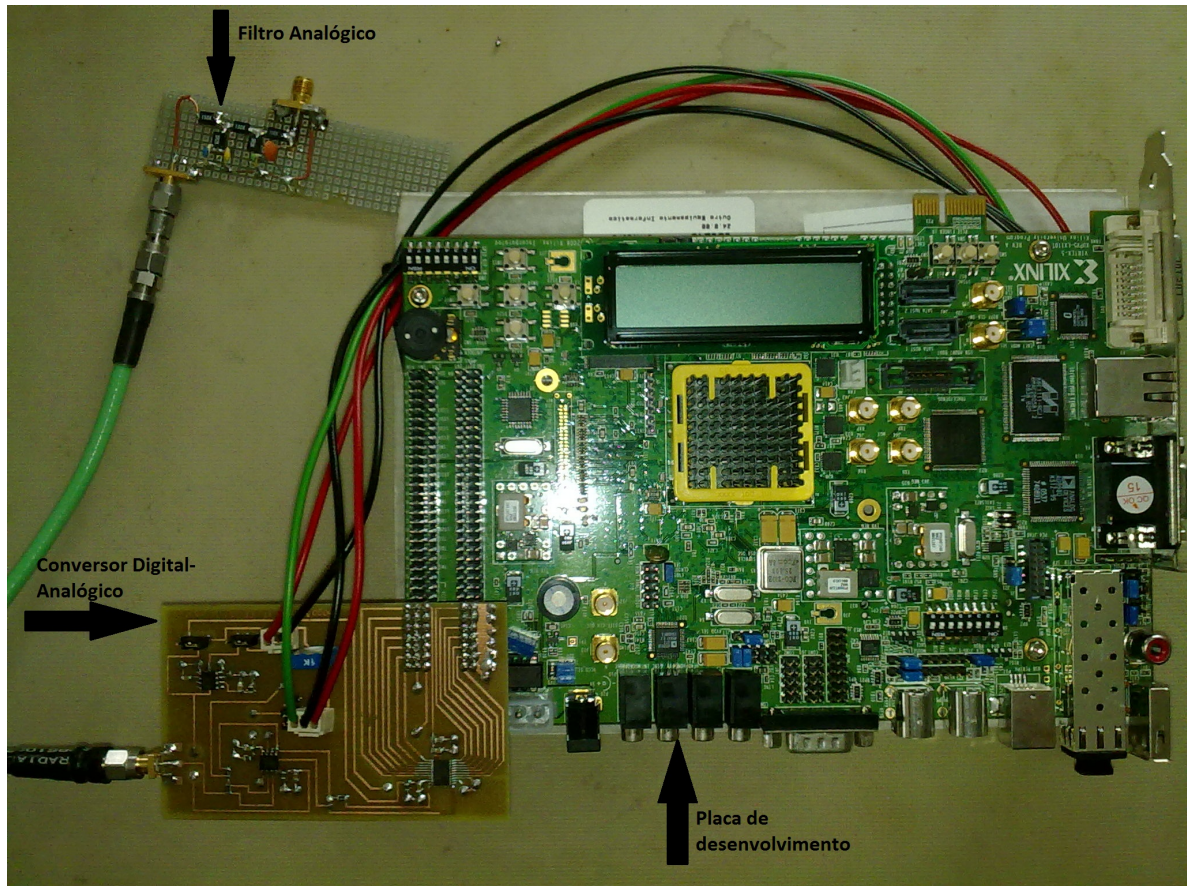


Figura 5.2: Placa de desenvolvimento, conversor Digital-Analógico e Filtro Analógico

Os testes efetuados irão recair nas mesmas especificações que se tiveram em conta nas simulações feitas, com a diferença que agora apenas se utiliza filtros digitais do tipo FIR. Observar-se-á qual o comportamento das máscaras espectrais quando se varia a frequência de amostragem e o número de bits de quantificação da DAC. E quando se utiliza o filtro analógico. A figura 5.3 ilustra o diagrama de fluxo utilizado para fazer os testes.

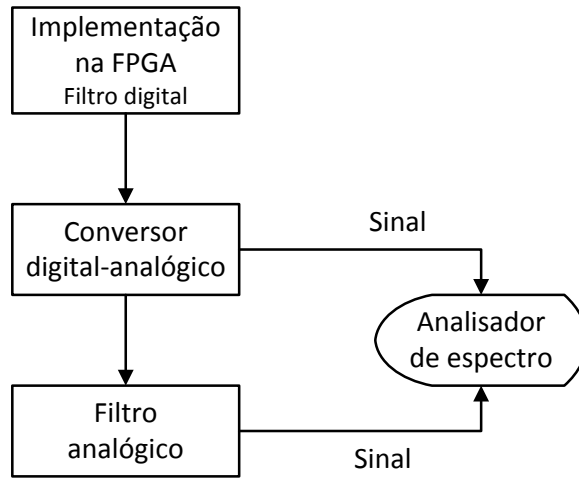


Figura 5.3: Diagrama de fluxo (medidas de radiação)

Fez-se vários testes distintos, por forma a se conseguir ter uma ideia geral do comportamento do sinal em função da variação dos parâmetros. Os testes incidiram sobre:

- Variação dos bits de quantificação da DAC (12, 10 e 8 bits), considerando um filtro FIR com 100 coeficientes e uma frequência de amostragem de 20 MHz;
- Variação da frequência de amostragem (20, 10 e 5 MHz), considerando um filtro FIR com 100 coeficientes e uma DAC com 12 bits de quantificação;
- Aplicação do filtro analógico, a seguir ao módulo que contém a DAC, com frequência de amostragem de 20 MHz e DAC com 12 bits de quantificação.

Para se fazer o teste de uma DAC de 8 e 10 bits aplicaram-se duas mudanças na implementação. Primeiro quantizou-se o valor à saída do filtro FIR em 8 e 10 bits. Posteriormente, à saída da FPGA, tomou-se os 2 ou 4 bits menos significativos como valores lógicos 0, visto a DAC inserida no módulo de conversão de dados ser de 12 bits.

Os passos seguidos, para se fazer o levantamento do espectro do sinal transmitido, são abaixo enunciados de forma sucinta. Contudo encontram-se pormenorizadamente caracterizados no documento [8], secção 9.9.

#	1	2	3	4	5	6	7	8	9
f_{offset} (MHz)	1	1,5	2	3	3,5	4	6	6,5	7
RWB (kHz)	30	100	100	100	100	30	30	100	100

Tabela 5.1: Frequências e RBW para o teste das máscaras espectrais do emissor

1. Aplicação do maior índice de modulação possível (90%);
2. Transmissão da sequência pseudo-aleatória;

3. Colocar o analisador de espectro no modo CW (*zero span mode operation*);
4. Selecionar uma das frequências (f_{offset}) da tabela 5.1. Caso sejam escolhidas as frequências 1, 4 ou 6 MHz deverá seguir-se para o passo seguinte, se não deverá passar-se para o décimo;
5. Define-se como frequência central do analisador $f_c = f_{offset} - RWB/2$, de acordo com a tabela 5.1;
6. Mede-se a potência (P_1) apresentada pelo analisador de espectro;
7. Define-se como frequência central do analisador $f_c = f_{offset} + RWB/2$, de acordo com a tabela 5.1;
8. Mede-se a potência (P_2) apresentada pelo analisador de espectro;
9. Calcula-se a potência total. Dado a potência (P_1 e P_2) se encontrar em dBm é necessário convertê-la para watt, somar e por fim convertê-la outra vez para dBm;
10. Define-se como frequência central do analisador $f_c = f_{offset} - RWB * (i - 2)$, de acordo com a tabela 5.1. O contador 'i' é posto a zero;
11. Mede-se a potência (P_1) apresentada pelo analisador de espectro;
12. Incrementa-se 'i' de '1' em '1' até atingir o valor '5'. À medida que se vai incrementado o contador mede-se as diferentes potências (P_2 , P_3 , P_4 e P_5);
13. Calcula-se a potência total. Dado a potência (P_1 , P_2 , P_3 , P_4 e P_5) se encontrar em dBm é necessário converte-la para watt, somar e por fim converte-la outra vez para dBm;
14. Repete-se todos os passos até se ter todas as potências representadas na tabela 2.1.

Variação do números de bits de quantificação da DAC

De maneira a verificar o efeito da quantificação introduzida pela DAC, fez-se a comparação entre os três casos distintos abaixo indicados. Para os três casos considerou-se um filtro FIR de ordem 100 e uma frequência de amostragem de 20 MHz. Os casos são:

- Dac com 12 bits de quantificação, frequência de amostragem de 20 MHz e filtro FIR de ordem 100;
- Dac com 10 bits de quantificação, frequência de amostragem de 20 MHz e filtro FIR de ordem 100;
- Dac com 8 bits de quantificação, frequência de amostragem de 20 MHz e filtro FIR de ordem 100.

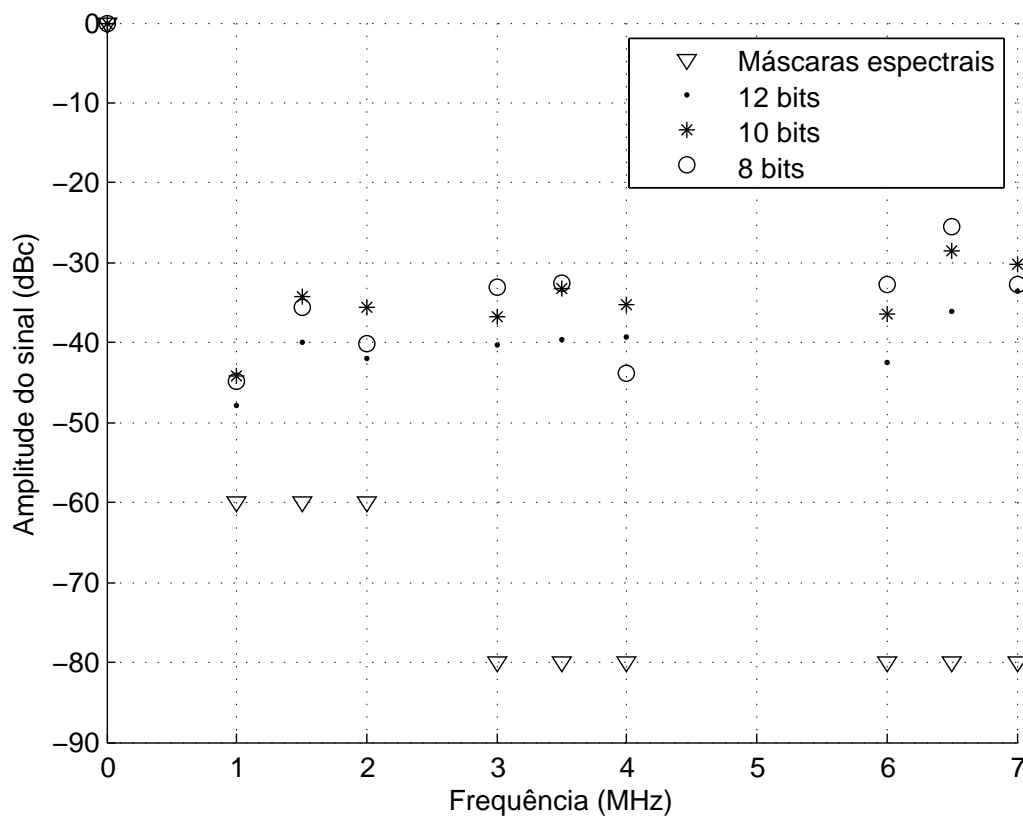


Figura 5.4: Variação do número de bits de quantificação da DAC

Conclui-se que quanto mais bits de quantificação se utilizarem maior é a atenuação obtida, em geral. De resto, este era o resultado esperado tendo em conta que quantos menos bits de quantização se utilizarem maior é o ruído de quantificação que se obterá, prejudicando o trabalho já efetuado pelo filtro FIR, colocado a montante.

Variação da frequência de amostragem

Verifica-se agora a existência ou não de alguma diferença na potência presente nas bandas em estudo, aquando da variação da frequência de amostragem. Neste caso as experiências feitas consistiram:

- Frequência de amostragem de 20 MHz, Dac com 12 bits de quantificação e filtro FIR de ordem 100;
- Frequência de amostragem de 10 MHz, Dac com 12 bits de quantificação e filtro FIR de ordem 100;
- Frequência de amostragem de 5 MHz, Dac com 12 bits de quantificação e filtro FIR de ordem 100;

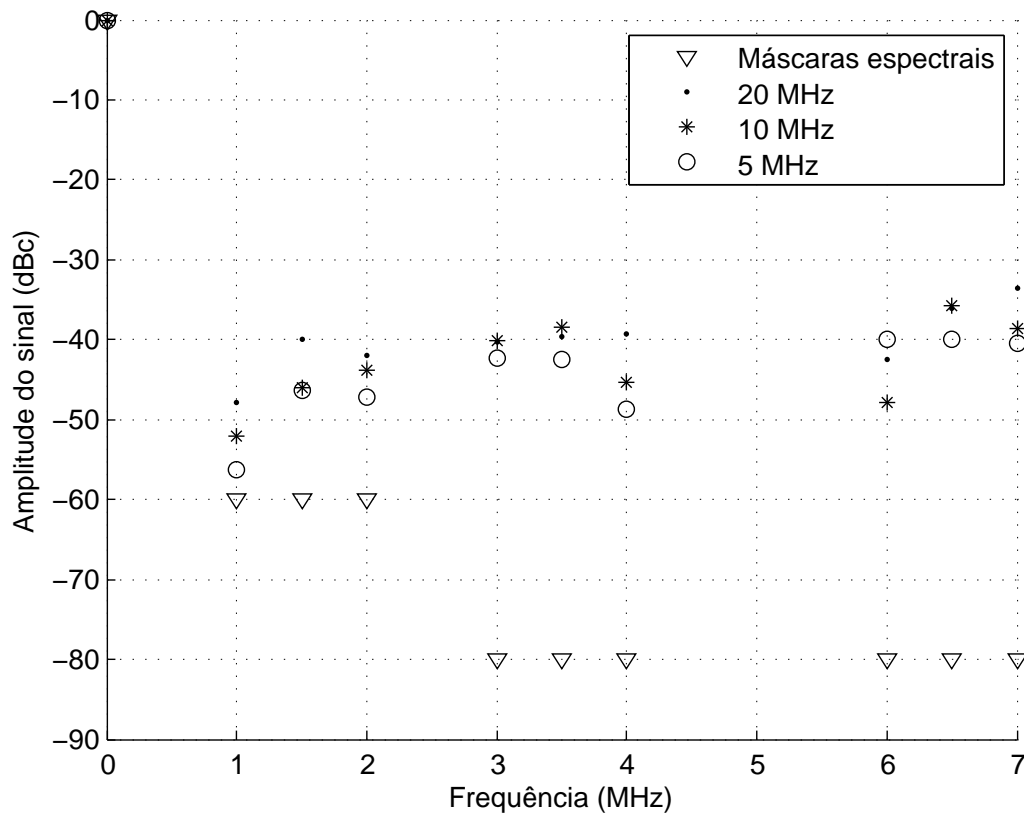


Figura 5.5: Variação da frequência de amostragem

Como já abordado aquando da simulação, a variação da frequência não deveria introduzir diferenças significativas no espectro do sinal, pois está-se a considerar frequências de amostragem superiores a 2x a largura de banda do sinal considerado. E efetivamente não existem diferenças expressivas. Todavia como se está a estudar uma largura de banda de 7 MHz, nos casos em que se utiliza as frequências de amostragem de 5 MHz e 10 MHz haverá *aliasing*. Contudo essa *aliasing* não será muito prejudicial pois corresponderá a zonas de espectro já atenuadas e não à banda onde se encontra a informação relativa ao sinal a ser enviado (0 - 1MHz).

No gráfico 5.5 observa-se efetivamente a repetição do sinal em torno dos 5 MHz, no caso de uma frequência de amostragem de 10 MHz, devido à *aliasing*.

Já quando a frequência de amostragem é de 5 MHz essa repetição não é tão evidente. Nota-se que o a potência do espectro tende a variar de forma semelhante em torno dos 2,5 MHz e 5 MHz contudo os níveis de potência são diferentes. Esta diferença parece indicar a presença de algum ruído no sinal, existente apenas aquando da aplicação desta frequência de amostragem. Mais à frente, aquando do estudo dos diagramas de olho do sinal, mais explicações serão dadas.

Aplicação do filtro analógico

Aplicou-se agora também ao sinal o filtro analógico já analisado. O sinal é presente à entrada do filtro depois de ter sido já filtrado por o filtro digital FIR e convertido em analógico pelo conversor digital-analógico. Considerou-se o caso:

- Filtro FIR com 100 coeficientes, frequência de amostragem = 20 MHz, DAC com 12 bits de quantificação e filtro analógico.

Fez-se a comparação com o sinal antes de passar pelo filtro analógico, de maneira a demonstrar a diferença entre a atenuação presente à entrada do filtro e a atenuação presente à sua saída.

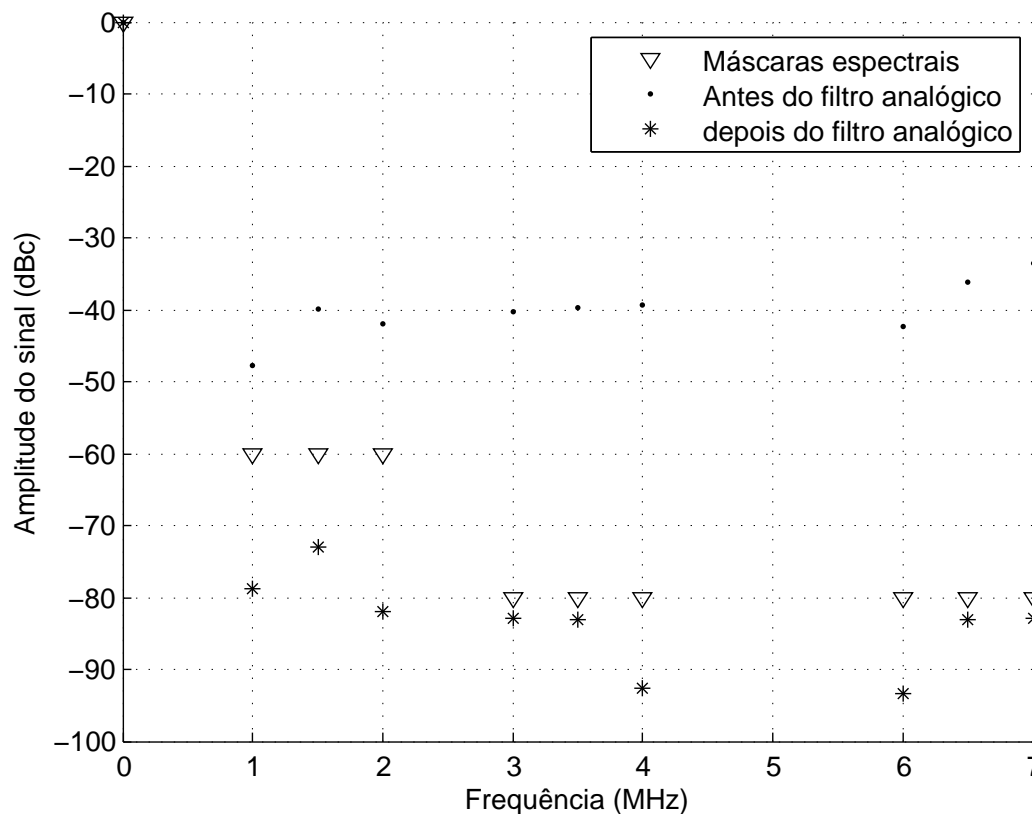


Figura 5.6: Aplicação do filtro analógico

O mais importante a retirar da figura 5.6 é que depois da aplicação do filtro analógico o sinal passa a cumprir as máscaras espectrais referidas na tabela 3.1. Porém deve-se referir que os níveis de potência foram de tal forma reduzidos que o chão de ruído do analisador de espectro era superior à potência do sinal presente nas bandas adjacentes em estudo, donde se conclui que o filtro analógico domina quase por completo a resposta do sistema.

Convém também salientar que ao efetuar-se estes testes houve a necessidade de se atenuar o sinal que se tinha à entrada do filtro analógico, pois observou-se que se tal não fosse feito o sinal era distorcido. Este problema devia-se à saturação das bobinas utilizadas. Colocou-se

por isso um atenuador de 6 dB à entrada do filtro analógico por forma a reduzir o nível do sinal. Houve também a necessidade de se colocar um atenuador de 6 dB à saída do mesmo para assegurar uma melhor adaptação entre a saída do filtro e a entrada do analisador de espectro. Esta atenuação não prejudica os resultados porque todo o sinal é atenuado e o que se mede é a diferença entre a potência presente no lobulo principal (0 até 1 MHz) e as bandas adjacentes.

5.3 Diagramas de olho

Os diagramas de olho são representações pelas quais se podem retirar várias conclusões acerca do sinal que se está a tratar. É possível observar se os sinais são muito ruidosos ou não, qual rapidez de transição entre níveis lógicos e se apresentam *overshoot* ou *undershoot*, por exemplo.

Estas representações consistem na amostragem repetida num osciloscópio do sinal que se está a tratar, usando como *trigger* (instante de amostragem do sinal) um relógio com frequência igual à taxa de transmissão do sinal amostrado.

A figura 5.7 em baixo apresentada é um exemplo de um diagrama de olho e as suas características principais. Esta figura foi retirada de maneira idêntica às que foram obtidas nos resultados abaixo.

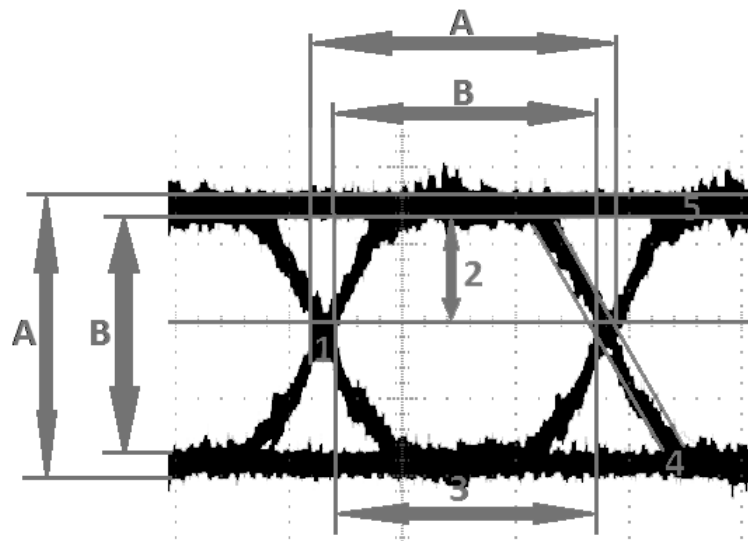


Figura 5.7: Características de um diagrama de olho

As características que se podem tirar de um sinal através da interpretação do seu diagrama de olho e que estão ilustradas na figura são:

1. Jitter, ou seja, a máxima distorção existente na transição entre níveis lógicos alto e baixo;
2. Imunidade do sinal ao ruído, quanto maior a abertura melhor;
3. Abertura do olho: espaço de tempo em que se pode fazer a amostragem do sinal com sucesso. O centro desta distância corresponde ao instante ideal;

4. O declive mostra a sensibilidade a erros de temporização. Quanto maior o declive melhor;
5. Quantidade de distorção no instante de amostragem. Está relacionada com a relação sinal-ruído do sinal.
6. Por fim através das dimensões representadas pelas letras A e B podem-se calcular figuras de mérito dos diagramas de olho e assim fazer-se uma comparação entre os vários diagramas obtidos. A equação utilizada para o cálculo é: $Abertura = \frac{2B}{A+B} * 100$.

Far-se-á o cálculo de duas aberturas distintas para cada diagrama de olho. A abertura na amplitude e a abertura no tempo, correspondendo às medições verticais e horizontais, respetivamente.

5.3.1 Testes e resultados

Retiraram-se alguns diagramas de olho do sinal transmitido, com frequências e o número de bits de quantificação distintos. Fez-se ainda a comparação do diagrama de olho do sinal antes e depois de passar pelo filtro analógico e com diferentes adaptações.

Estes resultados foram retirados usando um osciloscópio digital, aplicando a técnica de obtenção dos diagramas de olho já explicada. Os restantes equipamentos utilizados foram os mesmos dos testes anteriores. A figura 5.8 ilustra o modelo utilizado para se retirar os resultados. Especifica-se de onde são retirados os sinais e qual o canal de entrada de cada um deles no osciloscópio. De referir que não se terá dois sinais simultâneos no mesmo canal do osciloscópio, considerar-se-á um de cada vez.

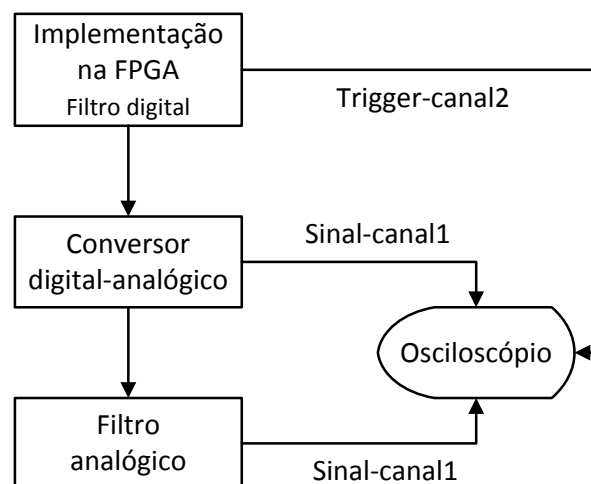


Figura 5.8: Diagrama de fluxo (diagramas de olho)

De forma a se conseguir ter a representação no osciloscópio das várias repetições do sinal aumentou-se a persistência do sinal. A representação foi depois passada para o computador sob a forma de imagem.

A figura 5.9 apresenta o diagrama de olho do sinal, com frequência de amostragem de 20 MHz, depois de passar por um filtro FIR de ordem 100 e por uma DAC de 12 bits.

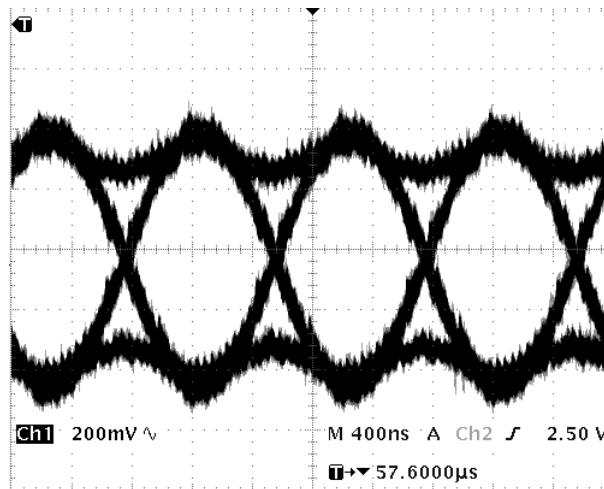


Figura 5.9: Diagrama de olho (sinal 20MHz, 12 bits)

É facilmente visível que o sinal é bastante bom pois o diagrama de olho apresenta uma boa abertura, pouco ruído e pouco jitter também. Tem-se uma abertura no tempo de 90% e na amplitude de 86%. De referir que as condições para se considerar que o sinal tem um bom diagrama de olho são uma abertura no tempo superior a 90% e uma abertura na amplitude superior a 85%. Logo pode-se considerar que o diagrama para o caso considerado é muito bom e estabelecer-se-á como referência para comparação com os diagramas seguintes.

Na figura 5.10 ilustra-se os diagramas de olho do sinal depois de passar por um filtro FIR de ordem 100 e por uma DAC de 12 bits, com frequências de amostragens diferentes (5MHz (a)) e 10MHz (b)).

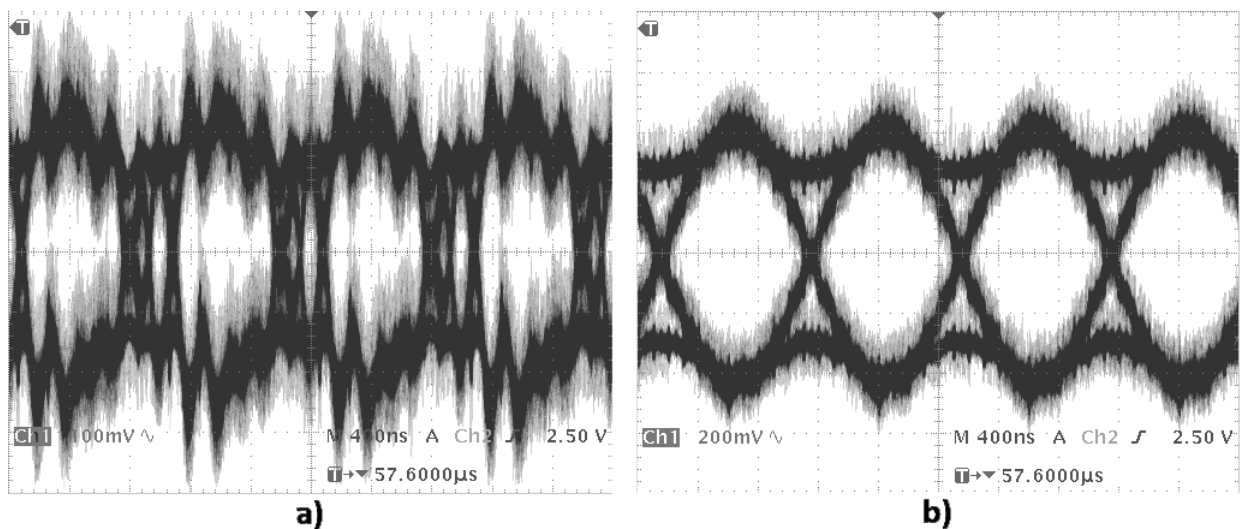


Figura 5.10: a)-Diagrama de olho (sinal 5MHz, 12 bits);
b)-Diagrama de olho (sinal 10MHz, 12 bits)

Para a figura 5.10 a), tanto a abertura na amplitude como no tempo têm o valor de 57%. Já para a figura 5.10 b), a abertura na amplitude é de 78% enquanto que no tempo é de 90%.

Infere-se por comparação com as aberturas da figura 5.9 que a diminuição da frequência de amostragem introduz um maior ruído no sinal. Denota-se que há uma grande degradação do diagrama de olho do sinal quando se diminui a frequência para 5 MHz, o que poderá impossibilitar o correto funcionamento do sistema. De referir que este *ripple* presente no diagrama de olho resultará das várias repetições do sinal ao longo do espectro, repetições essas que serão mais quanto menor for a frequência de amostragem do sinal.

Esta degradação/*ripple* já havia sido notada aquando dos resultados obtidos na variação da frequência de amostragem, na secção das máscaras espectrais. Agora, graças ao diagrama de olho, foi possível explicar e observar o porquê de tais resultados.

A ilustração 5.11, ao contrário das duas anteriores, refere-se a um sinal com frequência de amostragem de 20 MHz depois de passar por um filtro FIR de ordem 100 e por uma DAC de 8 bits (b)) e 10 bits (a)).

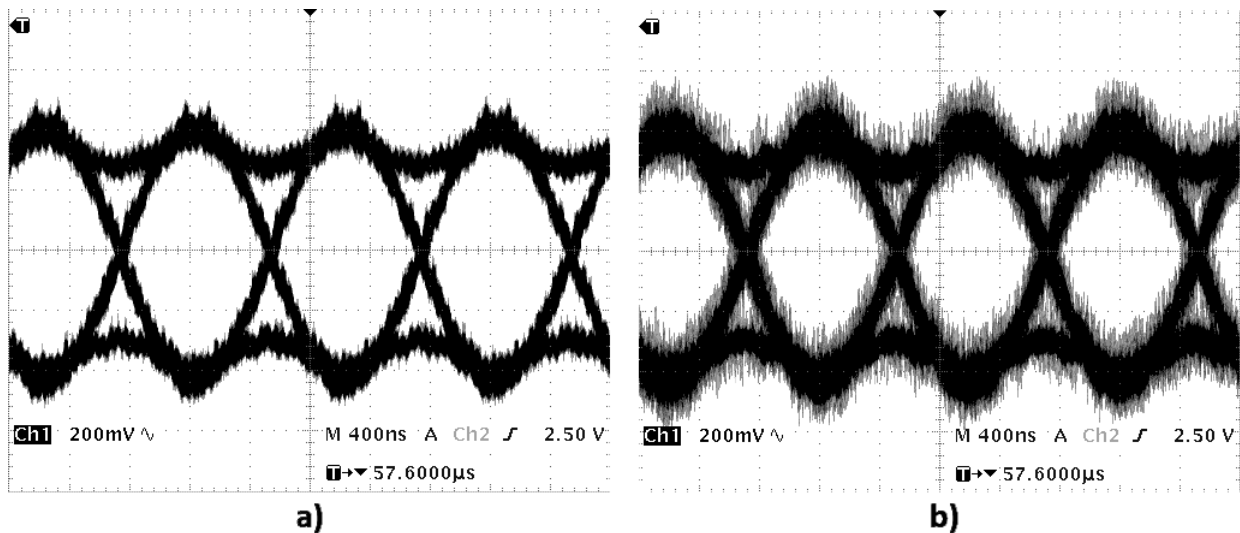


Figura 5.11: a)-Diagrama de olho (sinal 20MHz, 10 bits);

b)-Diagrama de olho (sinal 20MHz, 8 bits)

Mediu-se para a figura 5.11 a) uma abertura na amplitude de 86%, enquanto que a abertura no tempo foi de 90%. Para a figura 5.11 b) a abertura na amplitude medida foi de 80% ao passo que no tempo foi, à semelhança da figura anterior, de 90%. Através destas medições conclui-se, por comparação com a figura 5.9, que uma quantificação de 10 bits não introduz degradação no diagrama de olho do sinal. No entanto uma quantificação de 8 bits gera uma pequena degradação na abertura da amplitude do diagrama de olho do sinal. Há neste último caso um aumento do ruído, correspondente ao ruído de quantificação.

Por fim, a figura 5.12 ilustra os diagramas de olho do sinal, com frequência de amostragem de 20 MHz, depois de passar por um filtro FIR de ordem 100, por uma DAC de 12 bits e por o filtro analógico.

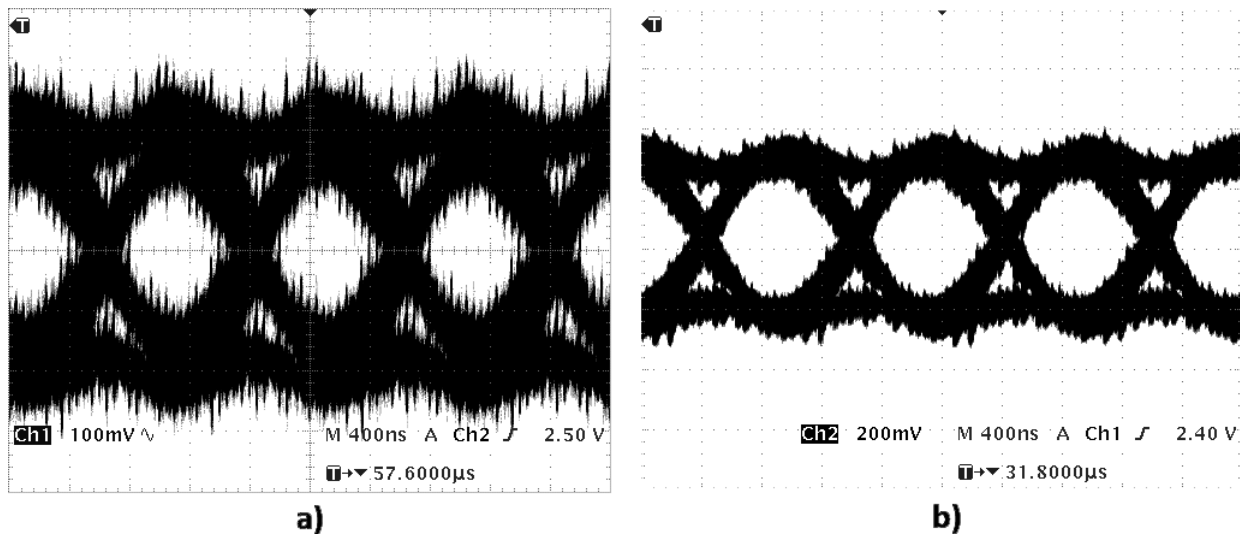


Figura 5.12: a)-Diagrama de olho (Filtro Analógico (6dB));
b)-Diagrama de olho (Filtro Analógico (10dB))

A diferença entre um e outro deve-se à adaptação existente entre a saída do filtro e a entrada do dispositivo seguinte (*mixer*). Ou seja, no caso da figura 5.12 (a)) colocou-se um atenuador de 6 dB, ao passo que no seguinte colocou-se um de 10 dB.

A abertura na amplitude para a figura 5.12 a) é de 58% e a abertura no tempo é de 70%. Para a figura 5.12 b) a abertura na amplitude é de 73% e abertura no tempo é de 80%. Por comparação com a figura 5.9 observa-se que a introdução do filtro analógico gera uma degradação no diagrama de olho do sinal, enquanto que a comparação das figuras 5.12 a) e b) permite inferir que há uma melhora no diagrama de olho do sinal com o aumento da adaptação.

Contudo uma melhor adaptação parece exigir uma maior atenuação, logo terá de se arranjar um meio termo de maneira a termos um sinal com um bom diagrama de olho e ao mesmo tempo uma boa potência. Todavia o ganho de 15 dB disponibilizado pela antena, tal como os ganhos dos andares de amplificação e do *mixer*, oferece uma certa liberdade para se poder atenuar o sinal à saída do filtro analógico.

5.4 Comunicação entre RSU e OBU

Por fim efetua-se uma tentativa de comunicação, em laboratório, entre o emissor e uma OBU comercial. Não se entra em conta com a distância de comunicação mínima e eventuais interferências que poderão existir em condições normais de funcionamento, ou seja, quando o emissor se encontra colocado em vias públicas. Apenas se quer assegurar que a implementação ao nível da FPGA e do conversor digital-analógico está bem feita, e por conseguinte se consegue efetuar com sucesso uma comunicação entre os dois dispositivos.

5.4.1 Testes e resultados

Por último, recorrendo a equipamentos já existentes, montou-se o transmissor completo (figura 5.13) e efetuou-se uma tentativa de comunicação entre a RSU e uma OBU comercial,

existente no laboratório. Neste caso enviou-se já uma mensagem válida do ponto de vista da OBU por forma a avaliar se esta responderia.

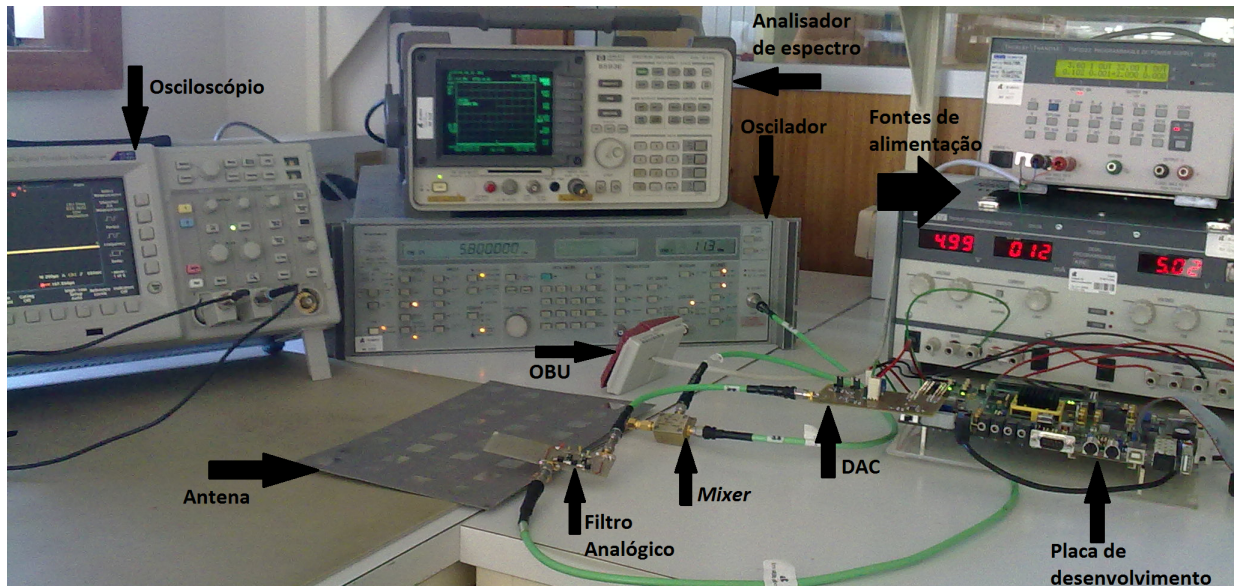


Figura 5.13: Teste de uma comunicação com uma OBU

De forma a testar o emissor construiu-se um sistema recorrendo a:

- FPGA;
- Conversor digital-analógico;
- Filtro analógico;
- *Mixer*;
- Oscilador 5,8 GHz;
- Antena com 15 dB de ganho.

Os equipamentos auxiliares para retirar resultados e alimentar o sistema foram:

- Duas fontes de alimentação para alimentar o conversor digital-analógico e a OBU;
- Osciloscópio;
- Analisador de espectro.

A figura 5.14 representa o diagrama de fluxo do sistema utilizado para retirar os resultados. É discriminado os locais onde foram captados os sinais apresentados nos resultados abaixo, tal como o encadeamento do tratamento do sinal pelo sistema.

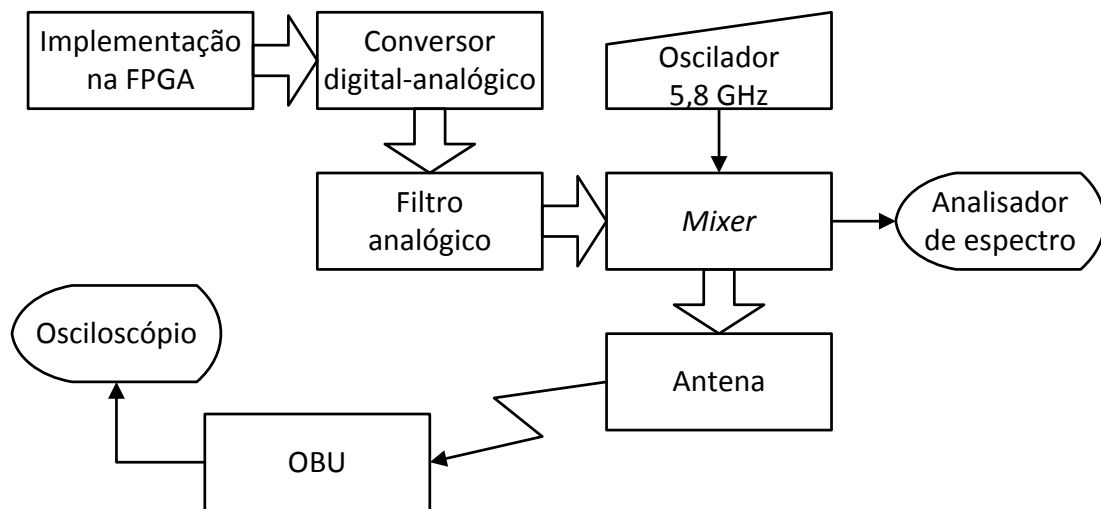


Figura 5.14: Diagrama de fluxo (comunicação entre RSU e OBU)

Na figura 5.15 (a) apresenta-se o espectro do sinal em torno dos 5,8 GHz. Os aspetos importantes a salientar é a presença da componente correspondente à portadora (5,8 GHz) e ausência de lobulos laterais. Isto é, observa-se que efetivamente a filtragem eliminou a maior parte da potência presente nas bandas adjacentes à banda principal. A PIRE medida foi de aproximadamente 11 dBm, ou seja, aproximadamente 12.6 mW. É um valor pequeno quando comparado ao valor máximo permitido (33 dBm), mas é já aceitável para os testes que serão efectuados.

Já a figura 5.15 (b) mostra o sinal que está a ser enviado pela RSU (em cima) e a recepção deste na OBU (em baixo). Comprova-se que o recetor da OBU está efetivamente a funcionar pois este está a desmodular corretamente o sinal que se está a enviar.

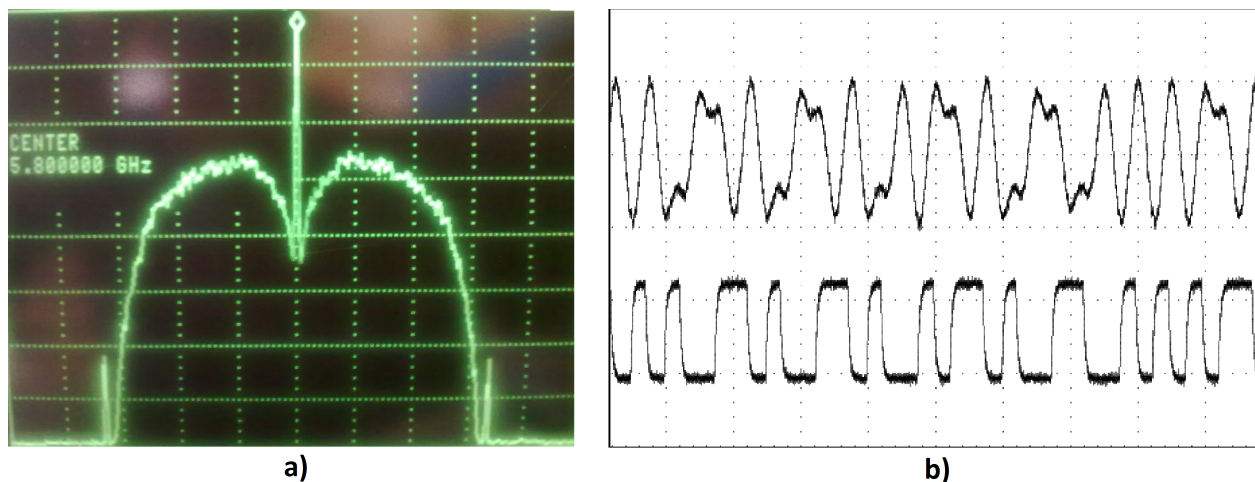


Figura 5.15: a)-Espectro do sinal a 5,8 GHz; b)-Sinal enviado pela RSU (em cima) e desmodulado pela OBU (em baixo)

Depois de se confirmar o funcionamento da OBU colocou-se a RSU a enviar uma mensagem

válida para ele e confirmou-se que efetivamente havia uma resposta da OBU. As várias partes correspondentes à trama de resposta enviada pela OBU encontram-se ilustradas nas figuras 5.16, 5.17, 5.18 e 5.19.

Os aspetos importantes a ter em conta em relação ao sinal de resposta enviado pela OBU são:

- Taxa de transmissão: 250 kbps;
- Frequência da subportadora: 1,5 MHz;
- Modulação aplicada: BPSK;
- Codificação de banda base: NRZI.

De notar que como a frequência da subportadora é 6 vezes superior à frequência de sinal, há que ter em atenção que cada bit de informação corresponde a seis transições entre o nível alto e o nível baixo da forma de onda. De referir também que neste tipo de codificação um zero lógico corresponde a uma inversão de fase do sinal ao passo que um nível lógico 1 implica manter a mesma fase.

O preâmbulo tem uma primeira parte de $32\ \mu\text{s}$ a $36\ \mu\text{s}$ só de sub-portadora (sem mudanças de fase). A segunda parte do preâmbulo é ilustrada na figura 5.16.

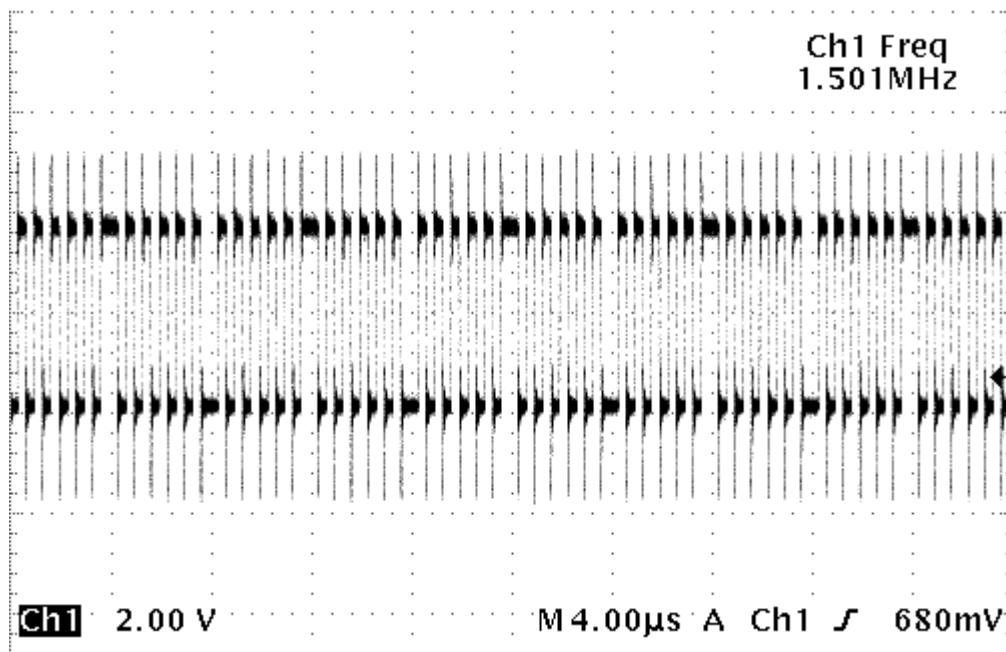


Figura 5.16: Preâmbulo da trama de resposta da OBU

Esta segunda parte do preâmbulo corresponde a 8 bits de nível lógico 0, tendo uma duração de $32\ \mu\text{s}$. O sinal apresenta transições no começo de cada novo bit, pois como já referido o nível lógico 0 corresponde a uma inversão de fase numa modulação BPSK.

Seguidamente são apresentadas as flags de início (5.17) e fim (5.18) da trama, que intercalam a parte da trama correspondente à informação.

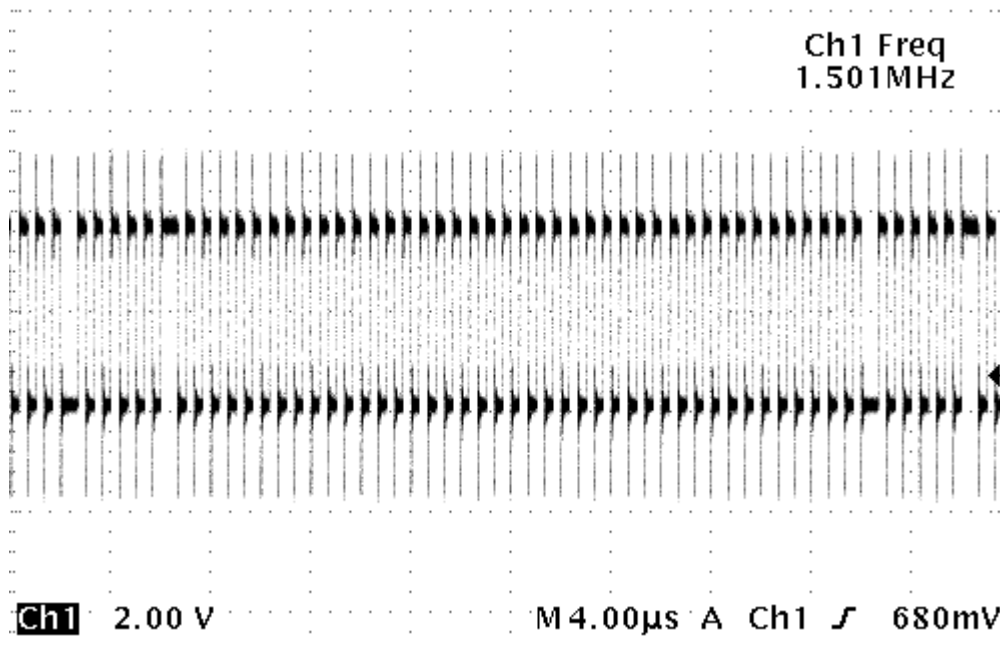


Figura 5.17: *Flag* de início da trama de resposta da OBU

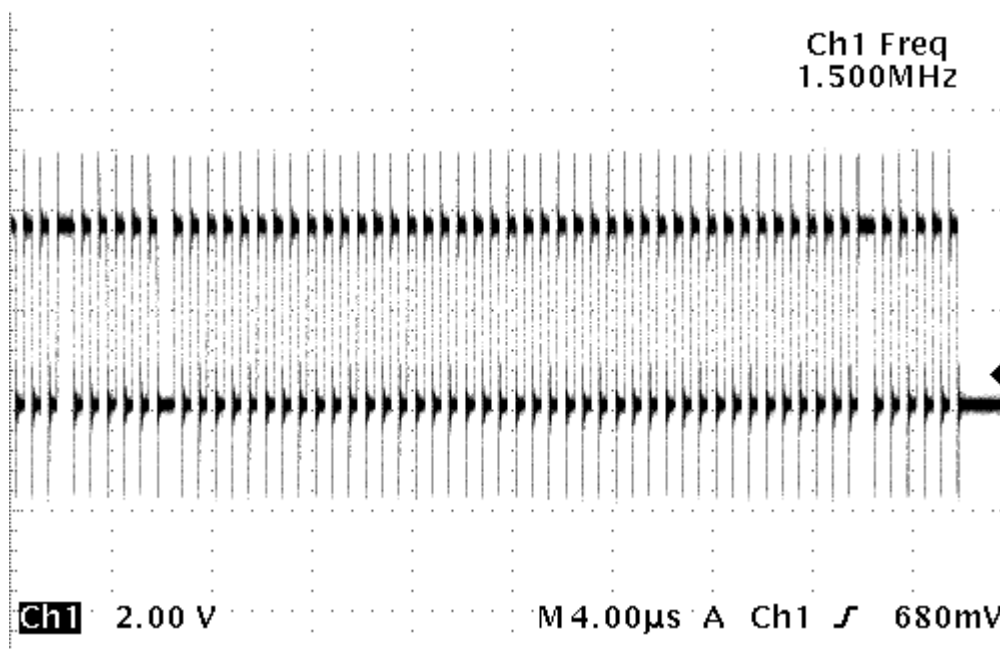


Figura 5.18: *Flag* de fim da trama de resposta da OBU

Tal como no caso da RSU as *flags* têm a forma mostrada na figura 2.8. Ou seja, como é visível nas figuras 5.17 e 5.18 há uma mudança de fase correspondente a um 0 seguida de seis 1's lógicos e acabando com outra mudança de fase (0 lógico). Têm a duração total de $32\mu s$.

Por fim a figura 5.19 ilustra um pequeno trecho da trama de informação.

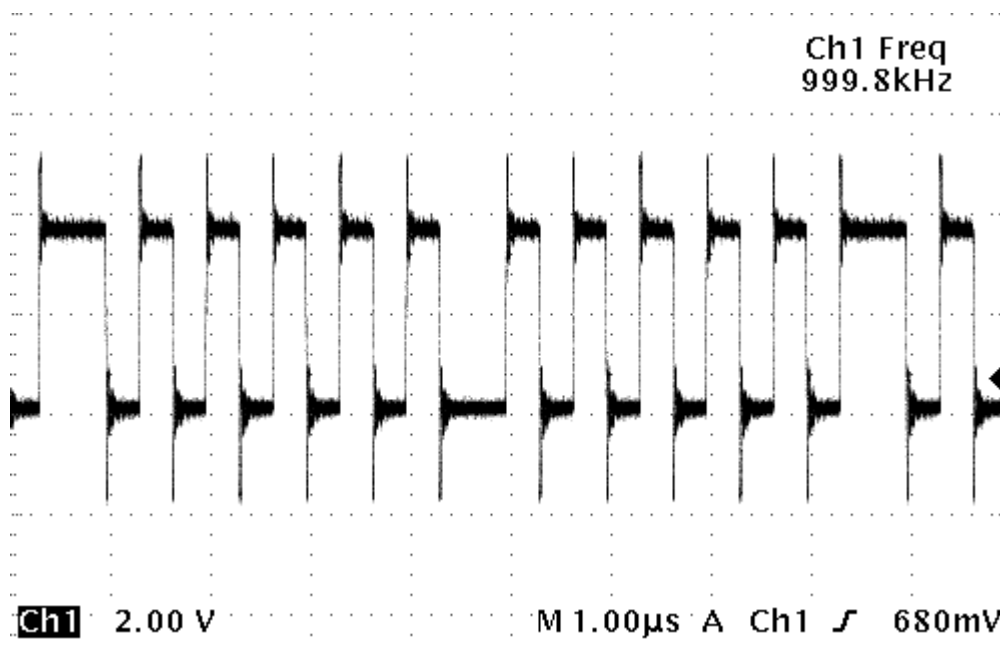


Figura 5.19: Exemplo de um trecho da trama de resposta da OBU

Neste trecho de informação pode-se observar mais pormenorizadamente o tempo de cada bit de informação. Consegue-se saber que os bits presentes neste trecho são 000, tendo em conta que cada bit demora $4\mu s$ e uma variação de fase corresponde a um 0 lógico.

Na codificação banda base NRZI quando se tem um 1 lógico mantém-se o nível anterior, ao passo de que quando se tem um 0 lógico há uma variação do nível do sinal. Se se entrar em linha de conta com este aspeto, pode-se afirmar que no trecho apresentado há pelo menos dois 1 lógicos, já que há três bits seguidos com o mesmo nível lógico.

Capítulo 6

Conclusões e trabalho futuro

6.1 Conclusões

Ao longo do trabalho foi feita a modelação da camada física e parte da camada MAC do sistema na FPGA, tal como a implementação da interface entre o meio digital e o meio analógico. Os passos seguidos tinham como meta o cumprimento dos seguintes objetivos:

1. Comunicar com um equipamento (OBU) comercial;
2. Cumprimento das máscaras espectrais enunciadas na norma EN 12253;
3. Construção do módulo de conversão do sinal de digital para analógico.

Se se tiver em conta os objetivos acima enunciados e os resultados obtidos pode-se afirmar que no seu todo o projeto foi completado com sucesso. Conseguiu-se interagir corretamente com um equipamento comercial e obter deste uma resposta corretamente formatada e as máscaras espectrais foram efetivamente cumpridas. Já no que ao conversor digital-analógico diz respeito constatou-se que este desempenhou bem o papel para o qual foi construído. Conclui-se, através das simulações e dos próprios resultados obtidos, que a DAC utilizada é adequada. Ou seja, será de todo preferível optar por uma que tenha no mínimo 12 bits de quantificação e uma frequência de amostragem superior a 14 MHz, de maneira garantir-se praticamente a inexistência de ruído de quantificação no sinal e evitar-se a presença de *aliasing* nas bandas em estudo do espectro. Logo a DAC escolhida é boa, pois encontra-se dentro destas especificações. Todavia deveria-se ter utilizado um outro filtro, específico para a reconstrução do sinal que se tem à saída do módulo da DAC. Filtro esse com uma frequência de corte mais elevada, servindo unicamente para eliminar as frequências mais elevadas do sinal, eliminando algum *overshoot* do sinal. Só seguidamente é que se introduziria o filtro analógico com a frequência de corte de 1 MHz de maneira a se cumprir as máscaras espectrais do sinal. Esta abordagem mais “sauve” produziria, porventura, um diagrama de olho mais aberto.

A adopção do *software defined radio* revelou ser uma boa escolha. Isto porque tornou o teste e estudo de diferentes abordagens mais simples e rápido, além de que novas funções ou alterações poderão ser facilmente introduzidas no sistema já implementado. O SDR providência uma grande liberdade ao sistema, o que é bastante valioso para qualquer projeto.

6.2 Trabalho futuro

Há aspetos que podem ser melhorados e/ou alterados, tais como:

- Filtragem aplicada: Uma das conclusões a que se chegou com a elaboração do trabalho foi de que o filtro analógico utilizado dominava quase por completo a resposta do sistema. Seria por isso pertinente o projeto de um filtro analógico menos complexo, menos “agressivo”, que proporcione diagramas de olho mais abertos. Também a elaboração de uma melhor filtragem a nível digital deverá ser alvo de estudo, de maneira a se conseguir atingir os limites de potência expressos na norma relativa à camada física, com uma maior facilidade. Uma das possibilidades será o estudo de possíveis implementações de filtros que não façam uso de multiplicadores, logo não serão tão restritivos em relação aos recursos disponíveis pela FPGA;
- Microcontrolador: No sistema implementado apenas se faz uso de uma FPGA, onde se implementou o tratamento imposto pela camada física do protocolo e o tratamento imposto pela camada MAC (*bitstuffing*, *flags*). Não houve incorporação de um microcontrolador, ou seja, tinha-se armazenada na FPGA uma mensagem para envio e estava-se continuamente a enviar, não estando implementadas parte da camada de ligação de dados e a camada de aplicação. Seria, por isso, interessante ter-se conseguido estabelecer uma interface entre um microcontrolador e a FPGA, de maneira a tornar o sistema mais completo. Contudo, é de referir que esse não era um objetivo do trabalho realizado.

Apêndices

Apêndice A

Código MatLab do emissor

O código apresentado corresponde ao usado na simulação do modelo, com o objetivo de se estudar o cumprimento das máscaras espectrais e o desempenho do conversor digital-analógico. Os passos efetuados foram:

- Criação da sequência a enviar;
- Codificação e Modulação da sequência;
- Filtragem recorrendo a filtros FIR, Butterworth (bw) e elípticos (el);
- Aplicação da DAC de 12, 10 e 8 bits;
- Cálculo da transformada de *Fourier*;
- Cálculo da potência em cada uma das bandas referidas pela norma.

```
%% constantes

% taxa de transmissão do sinal
bit_rate=500*10^3;

%taxa de transmissão da sequência codificada
bit_rate_cod = 1e6;

% varia entre 5 10 e 20 MHz
fa=20*10^6;

%níveis a aplicar na modulação
bin_min = 0.1;
bin_max = 1;

%números de bits da sequência
n_bits=1000;

%codigo
```

```

%% sequencia de bits a enviar
simb=zeros(1,n_bits);
x_p=randi([0,bin_max],1,n_bits);

% modulação
for i=1:length(x_p)
    if x_p(i)==0
        simb(i)=bin_min;
    else
        simb(i)=x_p(i);
    end
end

% codificação da sequencia -- codificação FM0
seq_cod=zeros(1,2*length(simb));
flag=0;
j=1;

for i=1:length(simb)
    if simb(i) == bin_max && flag == 1 && seq_cod(j-1)==bin_max
        seq_cod(j)=bin_min;
        seq_cod(j+1)=bin_min;
    end

    if simb(i) == bin_max && flag == 1 && seq_cod(j-1)==bin_min
        seq_cod(j)=bin_max;
        seq_cod(j+1)=bin_max;
    end

    if simb(i) == bin_min && flag == 1 && seq_cod(j-1)==bin_max
        seq_cod(j)=bin_min;
        seq_cod(j+1)=bin_max;
    end

    if simb(i) == bin_min && flag == 1 && seq_cod(j-1)==bin_min
        seq_cod(j)=bin_max;
        seq_cod(j+1)=bin_min;
    end

    if simb(i) == bin_max && flag == 0
        seq_cod(j)=bin_max;
        seq_cod(j+1)=bin_max;
        flag=1;
    end

    if simb(i) == bin_min && flag == 0
        seq_cod(j)=bin_max;

```

```

        seq_cod(j+1)=bin_min;
        flag=1;
    end
    j=j+2;
end

%% Aplicação da taxa de transmissão
for i=1:length(seq_cod)
    sinal(1+((i-1)*length(T_bit_cod)):i*length(T_bit_cod))=seq_cod(i);
end

for i=1:length(seq_trans)
    sinal_inic(1+((i-1)*length(T_bit)):i*length(T_bit))=seq_trans(i);
end

%% Representação do sinal sem codificação versus sinal com codificação
figure;
t=0:1/(fa):(length(sinal)*(1/fa))-(1/fa);
plot(t,((sinal)+1),'k');
hold on;
p=plot(t,sinal_inic);
set(p,'Color','black','LineWidth',3)
axis([4*10^-4 4.7*10^-4 -0.5 2.5])
legend('Sinal Codificado','Sinal Sem Codificação')
xlabel('Tempo (s)');
ylabel('Amplitude do sinal (V)');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% filtrar o sinal
%criação dos filtros passa baixo

% passa-baixo Butterworth
[n_butter, Wn_butter] = buttord(0.8e6/(fa/2), 1.4e6/(fa/2), 3, 50);
[B_butter,A_butter] = butter(n_butter, Wn_butter);

% passa-baixo elíptico
[n_ellip, Wp_ellip] = ellipord(0.9e6/(fa/2), 1.5e6/(fa/2), 3, 80);
[B_ellip,A_ellip] = ellip(n_ellip, 3, 80, Wp_ellip);

% passa-baixo FIR
h1=firpm(100, [0 0.86e6/(fa/2) 1.14e6/(fa/2) 1], [1 1 0 0]);

%Representação da resposta dos filtros
figure;
freqz(h1,1,1024,fa)
title('Filtro FIR 100 coeficientes')

```

```

figure;
freqz(B_butter,A_butter,1024,fa)
title('Filtro Butterworth')
figure;
freqz(B_ellip,A_ellip,1024,fa)
title('Filtro Elíptico')

% aplicação da filtragem do sinal com cascata de filtros
sinal_pb_bw=filter(B_butter, A_butter, sinal);
sinal_pb_bw_cas1=filter(B_butter, A_butter, sinal_pb_bw);
sinal_pb_bw_cas2=filter(B_butter, A_butter, sinal_pb_bw_cas1);
sinal_pb_bw_cas3=filter(B_butter, A_butter, sinal_pb_bw_cas2);

sinal_pb_el=filter(B_ellip, A_ellip, sinal);
sinal_pb_el_cas1=filter(B_ellip, A_ellip, sinal_pb_el);
sinal_pb_el_cas2=filter(B_ellip, A_ellip, sinal_pb_el_cas1);
sinal_pb_el_cas3=filter(B_ellip, A_ellip, sinal_pb_el_cas2);

sinal_pb_fir100=filter(h1, 1, sinal);
sinal_pb_fir100_cas1=filter(h1, 1, sinal_pb_fir100);
sinal_pb_fir100_cas2=filter(h1, 1, sinal_pb_fir100_cas1);
sinal_pb_fir100_cas3=filter(h1, 1, sinal_pb_fir100_cas2);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%cria-se um array de 2 dimensões para se armazenar os diferentes
%sinais (com e sem cascata de filtros)
%torna o programa mais simples
sinal_fir100(1,:) = sinal_pb_fir100;
sinal_fir100(2,:) = sinal_pb_fir100_cas1;
sinal_fir100(3,:) = sinal_pb_fir100_cas2;
sinal_fir100(4,:) = sinal_pb_fir100_cas3;

sinal_bw(1,:) = sinal_pb_bw;
sinal_bw(2,:) = sinal_pb_bw_cas1;
sinal_bw(3,:) = sinal_pb_bw_cas2;
sinal_bw(4,:) = sinal_pb_bw_cas3;

sinal_el(1,:) = sinal_pb_el;
sinal_el(2,:) = sinal_pb_el_cas1;
sinal_el(3,:) = sinal_pb_el_cas2;
sinal_el(4,:) = sinal_pb_el_cas3;

%faz-se todos os passos abaixo duas vezes uma considerando
% um filtro simples e outra aplicando a cascata de 2, 3, e 4 filtros
for v=1:4

%bits de quantificação - 12, 10 ou

```

```

5bits = 12;

%%%%%%%%      Aplicação da DAC      %%%%%%%%%

% sinal com filtro fir passa-baixo 100 coeficientes
niveis = 2^bits;
nivel_max1 = max(sinal_fir100(v,:));
nivel_min1 = min(sinal_fir100(v,:));
resolucao = (nivel_max1 - nivel_min1)/niveis;

nivel = nivel_min1;
for i=1:niveis+2
    niveis_q(i) = nivel;
    nivel = nivel + resolucao;
end

ruído_q=0;
m=1;
for i=1:length(sinal_fir100(v,:))
    flag = 0;
    for j=1:length(niveis_q)
        if niveis_q(j) > sinal_fir100(v,i) && flag == 0
            diff1 = abs(niveis_q(j-1)) - abs(sinal_fir100(v,i));
            diff2 = abs(niveis_q(j)) - abs(sinal_fir100(v,i));
            flag = 1;
            if diff1 < diff2
                sinal_fir100_q((div_fator*(i-1))+1:div_fator*i) = niveis_q(j-1);
                ruído_q(m) = diff1;
                m=m+1;
            else
                sinal_fir100_q((div_fator*(i-1))+1:div_fator*i) = niveis_q(j);
                ruído_q(m) = diff2;
                m=m+1;
            end
        end
    end
end
end

% sinal com filtro butterworth passa-baixo
niveis = 2^bits;
nivel_max2 = max(sinal_bw(v,:));
nivel_min2 = min(sinal_bw(v,:));
resolucao = (nivel_max2 - nivel_min2)/niveis;

nivel = nivel_min2;
niveis_q = 0;
for i=1:niveis+2

```

```

        niveis_q(i) = nivel;
        nivel = nivel + resolucao;
    end

    ruido_q=0;
    m=1;
    for i=1:length(sinal_bw(v,:))
        flag = 0;
        for j=1:length(niveis_q)
            if niveis_q(j) > sinal_bw(v,i) && flag == 0
                diff1 = abs(niveis_q(j-1)) - abs(sinal_bw(v,i));
                diff2 = abs(niveis_q(j)) - abs(sinal_bw(v,i));
                flag = 1;
                if diff1 < diff2
                    sinal_bw_q((div_fator*(i-1))+1:div_fator*i) = niveis_q(j-1);
                    ruido_q(m) = diff1;
                    m=m+1;
                else
                    sinal_bw_q((div_fator*(i-1))+1:div_fator*i) = niveis_q(j);
                    ruido_q(m) = diff2;
                    m=m+1;
                end
            end
        end
    end
end

%sinal com filtro eliptico passa-baixo
niveis = 2^bits;
nivel_max3 = max(sinal_el(v,:));
nivel_min3 = min(sinal_el(v,:));
resolucao = (nivel_max3 - nivel_min3)/niveis;

nivel = nivel_min3;
niveis_q = 0;
for i=1:niveis+2
    niveis_q(i) = nivel;
    nivel = nivel + resolucao;
end

ruido_q=0;
m=1;
for i=1:length(sinal_el(v,:))
    flag = 0;
    for j=1:length(niveis_q)
        if niveis_q(j) > sinal_el(v,i) && flag == 0
            diff1 = abs(niveis_q(j-1)) - abs(sinal_el(v,i));
            diff2 = abs(niveis_q(j)) - abs(sinal_el(v,i));

```



```

        flag = 1;
        if diff1 < diff2
            sinal_el_q((div_fator*(i-1))+1:div_fator*i) = niveis_q(j-1);
            ruido_q(m) = diff1;
            m=m+1;
        else
            sinal_el_q((div_fator*(i-1))+1:div_fator*i) = niveis_q(j);
            ruido_q(m) = diff2;
            m=m+1;
        end
    end
end
end

%Cálculos das FFT's dos sinais

%senal original
fft_b = abs(fft(sinal));

% sinal com filtro FIR de ordem 100 quantificado
fft_pb_fir100_q=abs(fft(sinal_fir100_q));
% sinal com filtro FIR de ordem 100
fft_pb_fir100 = abs(fft(sinal_fir100(1,:)));

% sinal com filtro Butterworth quantificado
fft_pb_bw_q=abs(fft(sinal_bw_q));
% sinal com filtro Butterworth
fft_pb_bw = abs(fft(sinal_bw(v,:)));

% sinal com filtro elíptico quantificado
fft_pb_el_q=abs(fft(sinal_el_q));
% sinal com filtro elíptico
fft_pb_el = abs(fft(sinal_el(v,:)));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
t=0:1/fa:(length(sinal)*(1/fa))-(1/fa);
N=length(t);

%vetor com as frequências no intervalo [0 fa]
F=(0:N-1)/N*fa;

%trunca-se os valores de maneira a conseguir-se
%corresponder os valores de F com valores inteiros
%como o bit_rate_cod
F=truncdig(F,5);

%cálculo das potências totais contida em cada um dos espectros

```

```

%- calcula-se para posteriormente se poder normalizar as
%potências de cada uma das bandas
p_fir100_q = (1/(N^2))*(sum(abs(fft_fir100_q).^2));
p_bw_q = (1/(N^2))*(sum(abs(fft_bw_q).^2));
p_el_q = (1/(N^2))*(sum(abs(fft_el_q).^2));
p_fir100 = (1/(N^2))*(sum(abs(fft_fir100).^2));
p_bw = (1/(N^2))*(sum(abs(fft_bw).^2));
p_el = (1/(N^2))*(sum(abs(fft_el).^2));
p = (1/(N^2))*(sum(abs(fft).^2));

% vetor com as larguras de banda correspondentes às
%frequências do vetor freqs
LB = [62.5*10^3 500*10^3 500*10^3 500*10^3 500*10^3
      500*10^3 500*10^3 62.5*10^3 62.5*10^3];

%vetor com as frequências aonde se deve calcular a potência
freqs = [1*10^6 1.5*10^6 2*10^6 3*10^6 3.5*10^6
         6.5*10^6 7*10^6 4*10^6 6*10^6];

%Escolha dos valores que interessam contidos no vetor F
%- neste caso quer-se só a parte correspondente ao lóbulo
%principal ou seja os valores nos intervalos [0 1MHz] e [fa-1MHz fa]
index1=find(F == bit_rate_cod);
index2=find(F == (fa-bit_rate_cod));
j=1;
for i=1 : index1
    LC_fir100_q(j)=fft_fir100_q(i);
    LC_bw_q(j)=fft_bw_q(i);
    LC_el_q(j)=fft_el_q(i);
    LC_fir100(j)=fft_fir100(i);
    LC_bw(j)=fft_bw(i);
    LC_el(j)=fft_el(i);
    LC(j)=fft(i);
    j=j+1;
end
for i=index2 : length(fft_fir100_q)
    LC_fir100_q(j)=fft_fir100_q(i);
    LC_bw_q(j)=fft_bw_q(i);
    LC_el_q(j)=fft_el_q(i);
    LC_fir100(j)=fft_fir100(i);
    LC_bw(j)=fft_bw(i);
    LC_el(j)=fft_el(i);
    LC(j)=fft(i);
    j=j+1;
end

%agora calcula-se a potência normalizada para cada

```

```

%um dos casos anteriores
dep_fir100_q(1)=(1/(N^2))*(sum(LC_fir100_q.^2));
dep_fir100_q_log(1)=10*log10(dep_fir100_q(1)/p_fir100_q);

dep_bw_q(1)=(1/(N^2))*(sum(LC_bw_q.^2));
dep_bw_q_log(1)=10*log10(dep_bw_q(1)/p_bw_q);

dep_el_q(1)=(1/(N^2))*(sum(LC_el_q.^2));
dep_el_q_log(1)=10*log10(dep_el_q(1)/p_el_q);

dep_fir100(1)=(1/(N^2))*(sum(LC_fir100.^2));
dep_fir100_log(1)=10*log10(dep_fir100(1)/p_fir100);

dep_fir60(1)=(1/(N^2))*(sum(LC_fir60.^2));
dep_fir60_log(1)=10*log10(dep_fir60(1)/p_fir60);

dep_fir30(1)=(1/(N^2))*(sum(LC_fir30.^2));
dep_fir30_log(1)=10*log10(dep_fir30(1)/p_fir30);

dep_fir20(1)=(1/(N^2))*(sum(LC_fir20.^2));
dep_fir20_log(1)=10*log10(dep_fir20(1)/p_fir20);

dep_bw(1)=(1/(N^2))*(sum(LC_bw.^2));
dep_bw_log(1)=10*log10(dep_bw(1)/p_bw);

dep_el(1)=(1/(N^2))*(sum(LC_el.^2));
dep_el_log(1)=10*log10(dep_el(1)/p_el);

dep(1)=(1/(N^2))*(sum(LC.^2));
dep_log(1)=10*log10(dep(1)/p);

%faz-se o mesmo que no caso anterior mas agora para cada uma
%das frequências contidas no vetor F considerando a largura
%de banda correspondente (vetor LB)
for k=1 : length(freqs)
    index=0;
    index=find(F <= freqs(k)-(LB(k)/2));
    index1=max(index);
    index=0;
    index=find(F <= freqs(k)+(LB(k)/2));
    index2=max(index);

    j=1;
    for i=index1 : index2
        CC_fir100_q(j)=fft_fir100_q(i);
        CC_bw_q(j)=fft_bw_q(i);
        CC_el_q(j)=fft_el_q(i);
    end
end

```

```

        CC_fir100(j)=fft_fir100(i);
        CC_bw(j)=fft_bw(i);
        CC_el(j)=fft_el(i);
        CC(j)=fft(i);
        j=j+1;
    end

    dep_fir100_q(k+1)=(1/(N^2))*(sum(CC_fir100_q.^2));
    dep_fir100_q_log(k+1)=10*log10(dep_fir100_q(k+1)/p_fir100_q);

    dep_bw_q(k+1)=(1/(N^2))*(sum(CC_bw_q.^2));
    dep_bw_q_log(k+1)=10*log10(dep_bw_q(k+1)/p_bw_q);

    dep_el_q(k+1)=(1/(N^2))*(sum(CC_el_q.^2));
    dep_el_q_log(k+1)=10*log10(dep_el_q(k+1)/p_el_q);

    dep_fir100(k+1)=(1/(N^2))*(sum(CC_fir100.^2));
    dep_fir100_log(k+1)=10*log10(dep_fir100(k+1)/p_fir100);

    dep_fir60(k+1)=(1/(N^2))*(sum(CC_fir60.^2));
    dep_fir60_log(k+1)=10*log10(dep_fir60(k+1)/p_fir60);

    dep_fir30(k+1)=(1/(N^2))*(sum(CC_fir30.^2));
    dep_fir30_log(k+1)=10*log10(dep_fir30(k+1)/p_fir30);

    dep_fir20(k+1)=(1/(N^2))*(sum(CC_fir20.^2));
    dep_fir20_log(k+1)=10*log10(dep_fir20(k+1)/p_fir20);

    dep_bw(k+1)=(1/(N^2))*(sum(CC_bw.^2));
    dep_bw_log(k+1)=10*log10(dep_bw(k+1)/p_bw);

    dep_el(k+1)=(1/(N^2))*(sum(CC_el.^2));
    dep_el_log(k+1)=10*log10(dep_el(k+1)/p_el);

    dep(k+1)=(1/(N^2))*(sum(CC.^2));
    dep_log(k+1)=10*log10(dep(k+1)/p);
end
end

```

Apêndice B

Front-End

O PCB construído apresenta uma DAC, dois amplificadores e um regulador de tensão. Teve-se o cuidado de seguir as indicações aconselhadas no *datasheet* do conversor digital-analógico, tais como inserir condensadores de acoplamento junto às alimentações dos circuitos integrados e fazer a ligação entre os planos de Terra (*ground*) digital e analógica o mais próximo possível da DAC.

Uma breve referência aos três principais componentes utilizados é apresentada a baixo:

- Conversor Digital-Analógico
 - Fabricante: ANALOG DEVICES;
 - Referência de fabricante: AD5445YRUZ;
 - Resolução (Bits): 12bit;
 - Frequência máximo de amostragem: 20.4MSPS;
 - Tipo de canal de entrada: Paralelo;
 - Interface dados: Paralelo;
 - Gama de alimentação analógica: 2.5V até 5.5V;
 - Corrente de alimentação: 0.4 μ A;
 - Preço/Unidade: 6,40€;
- Regulador de tensão
 - Fabricante: ANALOG DEVICES;
 - Referência de fabricante: ADR03ARZ;
 - Saída: 2,5V precisão;
 - Gama de alimentação analógica: 4.5V até 40V;
 - Pacote: SOIC 8;
 - Preço/Unidade: 4€;
- Amplificador
 - Fabricante: ANALOG DEVICES;
 - Referência de fabricante: AD8066ARZ;

- Alimentação simples e saída *rail-to-rail*
- Gama de alimentação analógica: 5V até 24V;
- Pacote: SOIC 8;
- Preço/Unidade: 6,6€

A abordagem na implementação do PCB foi sempre pensada de maneira a dar a maior liberdade possível ao sistema daí o uso de *jumpers* que permitem a comutação entre diferentes meios e níveis de alimentação. Ou seja, a DAC tanto pode aceitar níveis de tensão, para os sinais digitais, de 3,3 volt como 5 volt, além de se permitir o uso de uma alimentação externa. O *software* utilizado para a elaboração do PCB foi o *eagle*.

A figura B.1 ilustra o esquemático do circuito impresso no PCB. É possível observar todos os componentes usados e a sua interligação no interior do módulo. Este esquema é semelhante ao circuito representado na figura 4.6, mas mais pormenorizado. De referir que os *PADs* que se vêem no esquema correspondem aos encaixes que irão ser usados para interligar o módulo aos *expansion headers* da placa de desenvolvimento.

As figuras B.3 e B.2 representam o *Bottom* e o *Top* do PCB construído. No *top* pode-se observar como estão dispostos os vários componentes ilustrados no esquemático. Por forma a não “abrir” o plano de massa (*bottom*), optou-se por fazer algumas ligações recorrendo a fios, ao invés de pistas impressas no PCB. Estas ligações correspondem apenas a ligações de alimentações, não sendo por isso muito críticas.

No *bottom* pode-se observar o pormenor da interligação entre o plano de massa digital e o plano de massa analógica, tal como é aconselhado pelo fabricante no *datasheet* da DAC.

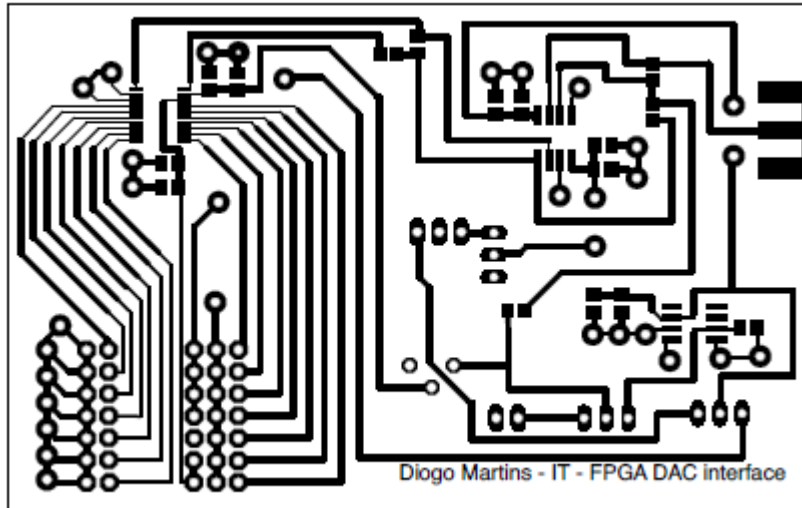


Figura B.2: PCB construído (top)

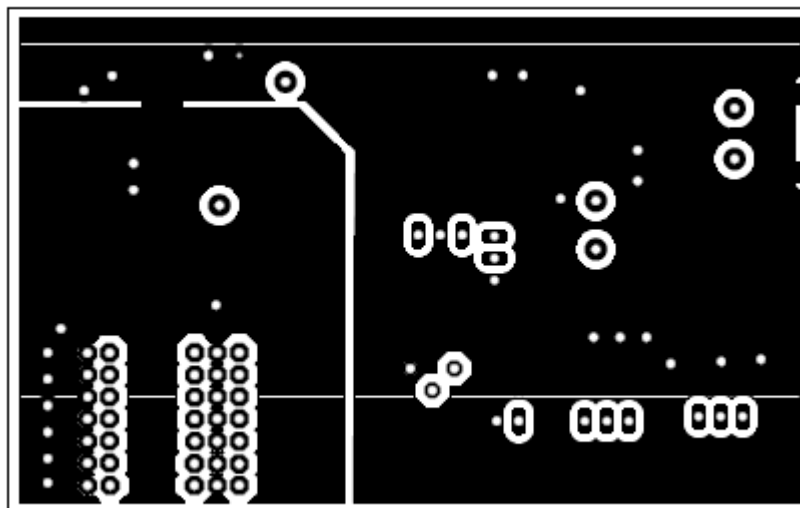


Figura B.3: PCB construído (bottom)

Bibliografia

- [1] Hyunseo Oh, Chungil Yae, Donghyon Ahn, and Hanberg Cho. 5.8 ghz dsrc packet communication system for ITS services. In *Vehicular Technology Conference, 1999. VTC 1999 - Fall. IEEE VTS 50th*, 1999.
- [2] Xiaoqing Zeng, Chenliang Tao, and Zhi Chen. The application of DSRC technology in intelligent transportation system. In *Wireless Mobile and Computing (CCWMC 2009), IET International Communication Conference on*, pages 265 – 268, 2009.
- [3] S. Winder. *Analog and digital filter design*. EDN series for design engineers. Newnes, 2002.
- [4] C. Bergstrom, S. Chuprun, S. Gifford, and G. Maalouli. Software defined radio (SDR) special military applications. In *MILCOM 2002. Proceedings*, volume 1, pages 383 – 388 vol.1, oct. 2002.
- [5] B. Tarver, E. Christensen, and A. Miller. Software defined radios (SDR) platform and application programming interfaces (API). In *Military Communications Conference, 2001. MILCOM 2001. Communications for Network-Centric Operations: Creating the Information Force. IEEE*, volume 1, pages 153 – 157 vol.1, 2001.
- [6] http://www.xilinx.com/support/documentation/ip_documentation/fir_compiler_ds534.pdf, 2011.
- [7] http://www.analog.com/static/imported-files/data_sheets/AD5424_5433_5445.pdf, 2011.
- [8] ETSI EN 300 674-1 V1.2.1 (2004-08). http://www.etsi.org/deliver/etsi_en/300600_300699/30067401/01.02.01_60/en_30067401v010201p.pdf, 2004.
- [9] Alex. Doboli and Edward H. Currie. *Introduction to Mixed-Signal, Embedded Design*. Springer, 2011.
- [10] W.H.W. Tuttlebee. *Software defined radio: baseband technologies for 3G handsets and basestations*. Wiley series in software radio. J. Wiley, 2004.
- [11] C. Britton Rorabaugh. *Digital Filter Designer's Handbook*. McGraw-Hill Education - Europe, 1993.
- [12] Farid Dowla. *Handbook of RF and Wireless Technologies*. Elsevier, 2004.

- [13] Ricardo Matos Abreu. Projecto de um receptor de microondas para uso em sistemas DSRC/RTTT. Master's thesis, Universidade de Aveiro, Departamento de Electrónica, Telecomunicações e Informática, 2007.
- [14] J.H. Reed. *Software radio: a modern approach to radio engineering*. Prentice Hall, 2002.
- [15] C. Cseh. Architecture of the dedicated short-range communications DSRC protocol. In *Vehicular Technology Conference, 1998. VTC 98. 48th IEEE*, volume 3, pages 2095–2099 vol.3, May 1998.
- [16] <http://www.etsi.org/WebSite/homepage.aspx>, 2011.
- [17] Road transport and traffic telematics - Dedicated short-range communication - Physical layer using microwave at 5,8 GHz, 1997.